

基于伴随方法的机翼气动弹性优化设计

申伟宏

院（系）： 航天学院 专 业： 航空宇航科学与技术
学 号： 1141820225 指导教师： 李顺利教授

2018 年 6 月 19 日

哈爾濱工業大學

畢業設計（論文）

題 目 基于伴随方法的

机翼气动弹性优化设计

专 业 航空宇航科学与技术

学 号 1141820225

学 生 申伟宏

指 导 教 师 李顺利教授

答 辩 日 期 2018 年 6 月 19 日

摘 要

本文以大展弦比机翼的气动和结构特性为研究对象，通过伴随方法求解性能指标对设计变量的导数，高效率完成了对机翼设计的优化。

气动弹性需要综合考虑结构和气动两个方面；在结构力学方面，本文以求解二维泊松方程为例利用有限元方法的基本理论给出求解过程，推导了弯曲单元和扭转单元的刚度矩阵，对比了线性矩阵方程组的常用求解算法的效率和精度，并完成有限元求解器的程序；在气动力方面，本文研究了片条理论，升力线方法和涡格法的基本理论并编写了相关求解器程序；对于静气动弹性问题，研究了几种求解耦合状态变量的迭代算法，并使用非线性分块高斯-赛德尔迭代方法编写求解静气动弹性状态变量的程序；研究了包括有限差分法，复数差分法和自动微分法在内的若干种常见的计算机求导数的算法，并具体研究伴随方法求解导数的好处，从一般性的伴随方法的方程推导到有限元的伴随方法方程，根据问题的解析解与有限差分法求导数的效率和精度进行对比，证明了伴随方法的优点：高效而且精确。最后，提出一个弹性机翼的优化问题，并提出约束条件为应力不超过屈服应力，针对弹性机翼的优化问题，推导流固耦合的伴随方程，通过伴随方法计算导数并与有限差分法计算导数的效率进行对比，并利用得到的导数信息通过移动渐近线算法对机翼的性能指标进行优化，最终得到优化后的机翼的结构设计，高效得到了机翼应力尽可能接近屈服应力、重量尽可能轻而产生的升力尽可能大的优化结果。

关键词：伴随方法；有限元；片条理论；气动弹性；优化设计

Abstract

In this article, the aerodynamic and structural characteristics of high-aspect-ratio wings are taken as the research object. The adjoint method is used to solve the derivative of the performance index to the design variables, which helps the optimization of the wing design to be completed with high efficiency.

The aeroelastic design needs to consider both structural and aerodynamic disciplines. In terms of structural mechanics, this paper takes the 2D Poisson equation as an example to study the basic theory of the finite element method and then deduces the stiffness matrix of the bending unit and the torsion unit. A comparison of different commonly used algorithms' efficiency and precision for matrix equations is also presented here, and the program for the finite element solver is completed. In the aspect of aerodynamics, this paper studies the strip theory, the lifting line method and the basic theory of vortex lattice method and writes the related solver program. For the static aeroelastic problem, several iterative algorithms for solving coupled state variables are studied, and a program for solving static aeroelastic state variables is compiled using a nonlinear block Gauss-Seidel method. This article also studies several ways to calculate derivatives on computer including finite difference method, complex-step derivative and automatic differentiation, in which we focus on the benefits of the adjoint method. General adjoint equations are derived and then the adjoint equations of the finite element method is get. The comparison of the efficiency and precision of the derivative between adjoint method and the finite difference method is demonstrated with known analytical solution of the problem. Efficiency and precision can be seen undoubtedly as advantages of adjoint method. Finally, an optimization problem of the elastic wing is proposed and the constraint condition is that the stress cannot exceed the yield stress. For the optimization problem of the elastic wing, the adjoint equations of the fluid-structure coupling is deduced. Calculating the derivative by the adjoint method is shown far more efficiently than calculating the derivative with the finite difference method. The derivative's performance index is optimized by the moving asymptotic algorithm using the derivative information obtained by adjoint method. The optimized structure of wing design is finally obtained, and the wing gets to lighter, stronger,

lifter with the stress obtained as close as possible to the yield stress.

Keywords: Adjoint Method, Finite Element Method, Strip Theory, Aeroelastic, Optimized Design

物理量名称及符号表

表 1 国际单位制中具有专门名称的导出单位

量的名称	单位名称	单位符号	其它表示实例
频率	赫 [兹]	Hz	s^{-1}

目 录

摘 要	I
ABSTRACT	II
物理量名称及符号表	IV
第 1 章 绪论	1
1.1 课题背景及研究的目的和意义	1
1.2 相关理论的发展概况	1
1.3 本文的主要研究内容	3
第 2 章 有限元方法	4
2.1 引言	4
2.2 有限元分析的方法论	4
2.3 有限元求解弯曲单元的模型	6
2.4 有限元求解具有弯曲和扭转的模型	7
2.5 有限元方程的求解	8
2.6 小结	10
第 3 章 气动计算方法	11
3.1 引言	11
3.2 片条理论	11
3.3 升力线法	12
3.4 涡格法	13
3.5 小结	14
第 4 章 流固耦合方程的求解	15
4.1 引言	15
4.2 流固耦合的残差方程	15
4.3 非线性分块高斯-赛德尔	17
4.4 非线性分块雅各比	17
4.5 牛顿-科里洛夫方法	17
4.6 近似牛顿-科里洛夫方法	18

4.7 静气动弹性机翼升力重新分布求解实例	18
4.8 小结	19
第 5 章 伴随方法	20
5.1 引言	20
5.2 梯度的计算方法	20
5.2.1 有限差分法	20
5.2.2 复数差分法	20
5.2.3 自动微分法	21
5.3 伴随方法与机翼优化	21
5.4 伴随方法与有限元模型的结合	22
5.4.1 简化模型	22
5.4.2 复杂模型	23
5.5 伴随方法与气动力方程结合	24
5.6 具体指标的梯度计算	25
5.6.1 应力的梯度	25
5.6.2 KS 函数的梯度	25
5.7 计算效率对比	26
5.8 小结	27
第 6 章 机翼优化实例	28
6.1 引言	28
6.2 流固耦合方程中梯度的计算	28
6.3 机翼的优化结果	30
6.4 小结	33
结 论	34
参考文献	35
哈尔滨工业大学本科毕业设计（论文）原创性声明	37
致 谢	38
附录 1 外文资料的调研阅读报告或书面翻译	39
1.1 引言	39
1.1.1 MDO 架构	39
1.1.2 动力	40

1.1.3 架构实现	41
1.1.4 气动结构模型	43
1.2 耦合后优化敏感度	44
1.2.1 气动残差	45
1.2.2 结构残差	45
1.2.3 结构优化残差	45
1.2.4 气动灵敏度	47
1.2.5 结构分析灵敏度	47
1.2.6 结构优化灵敏度	48
1.3 结果和讨论	48
1.4 结论	49
附录 2 外文资料原文	50
2.1 Introduction	50
2.1.1 MDO architecture	50
2.1.2 Motivation	51
2.1.3 Architecture implementation	52
2.1.4 Aero-structural model	54
2.2 Coupled post-optimality sensitivities (CPOS)	55
2.2.1 Aerodynamic residuals	56
2.2.2 Structural residuals	57
2.2.3 Structural optimization residuals	57
2.2.4 Aerodynamic sensitivities	59
2.2.5 Structural analysis sensitivities	60
2.2.6 Structural optimization sensitivities	60
2.3 Results and discussion	61
2.4 Conclusions	62

Contents

Abstract (In Chinese)	I
Abstract (In English)	II
List of physical quantity and symbol	IV
Chapter 1 Introduction	1
1.1 Background, Purpose and Significance of the Study	1
1.2 Development of Related Theories	1
1.3 The Main Research Content of This Article	3
Chapter 2 Basic Theory about Finite Element Method	4
2.1 Introduction	4
2.2 Methodology about FEM	4
2.3 Flexure Elements: The Direct Stiffness Method	6
2.4 A General Three-Dimensional Beam Element with Torsion	7
2.5 Method to Solve the FEM Linear Equation	8
2.6 Summary	10
Chapter 3 Aerodynamics Calculation Method	11
3.1 Introduction	11
3.2 Strip Theory	11
3.3 Lifting-Line Theory	12
3.4 Vortex Panel Method	13
3.5 Summary	14
Chapter 4 Method to Solve Aeroelasticity Coupled Equation	15
4.1 Introduction	15
4.2 Aero-Structural Coupled Residual Equations	15
4.3 Nonlinear Block Gauss-Seidel	17
4.4 Nonlinear Block Jacobi	17
4.5 Newton-Krylov Method	17
4.6 Approximate Newton-Krylov Method	18

4.7 A Static Aeroelastic Solution of a Lift Re-distribution Wing	18
4.8 Summary	19
Chapter 5 Adjoint Method	20
5.1 Introduction	20
5.2 Several Ways to Get Gradients.....	20
5.2.1 Finite Difference Method.....	20
5.2.2 Complex-Step Derivative Approximation.....	20
5.2.3 Automatic Differentiation	21
5.3 Adjoint Method and Wing Optimization	21
5.4 Adjoint Method Combined with FEM.....	22
5.4.1 A Simple Model	22
5.4.2 A Complex Model.....	23
5.5 Adjoint Method Combined with Aerodynamics Equations	24
5.6 Gradients on Specific Indexes	25
5.6.1 Gradients of Stress	25
5.6.2 Gradients of Ks Function.....	25
5.7 Computation Cost Comparison.....	26
5.8 Summary	27
Chapter 6 An Example of Wing Optimization Design	28
6.1 Introduction	28
6.2 Calculating Gradients in Aero-Structural Coupled Equations	28
6.3 Results of An Optimized Wing	30
6.4 Summary	33
Conclusions	34
References	35
Acknowledgements	38

第 1 章 绪论

1.1 课题背景及研究的目的和意义

在今天，复杂航空器的设计依然是一个艰巨的挑战，因为涉及到流体、结构与热耦合效应。尽管在设计中，直觉、经验与工程技能占据主要地位，数值模拟仍然是一种前景广阔的手段，因为这样可以缩短设计周期。

如今，随着科学技术的进步，人们对于空间的探索需求越来越大，在航空器方面太阳能飞机、大展弦比无人机(图1-1)、大型运输机等更多地出现在人们的视野中，在人们的生产生活中发挥出越来越大的作用。这类飞行器往往会采用高强度轻质材料且要求材料有较薄的成型潜力，并采用高效率的结构形式，例如桁架翼肋/机身、蜂窝/泡沫夹心结构、薄壁盒式/管型梁等。对于这些不同用途的飞机，我们会希望飞机的重量更轻、载重量更大、巡航速度更合理、气动弹性的发散速度更大、颤振频率更高等等，这就需要在飞行器的设计过程中对许多设计变量进行平衡以达到性能指标的最优化。



图 1-1 大展弦比无人机 RQ-4A
Fig. 1-1 High Aspect-Ratio UAV RQ-4A

1.2 相关理论的发展概况

通常，一个设计的优化问题由一组设计变量，一个需要优化的性能指标以及需要满足的约束条件组成。在气动弹性优化问题中，约束条件通常由耦合的偏微分方程组成。在优化问题中，由于耦合带来的巨大计算量，优化问题通常只关注结构或气动的某一方面而忽略另一个方面，如将气动外形视为刚体而对气动外形进行优化，例如陈颂利用梯度方法对气动外形进行优化^[1]，或者忽略结构变形造成的

气动载荷变化而对结构进行优化，例如茅志颖利用渐进优化方法对附加阻尼结构对结构动力学设计进行优化^[2]。这样处理可以减少运算量，但会带来一定的不精确性。

气动弹性优化通常是解决结构重量、结构强度、升阻特性之间的矛盾与平衡问题。在气动弹性优化问题中，设计变量通常包括扭转角、前缘下垂与后缘下垂、结构厚度、复合材料铺层的方向等。首先，结构重量直接决定所需要的升力，而升力又反作用于诱导阻力。其次，气动载荷将影响结构的变形，因而改变了气动外形。因此，为了得到一个收敛的气动-结构状态，通常需要若干步迭代。一个目标函数的表达式可以是：

$$R = \frac{V}{c} \frac{L}{D} \ln \frac{W_i}{W_f} \quad (1-1)$$

其中 V 为巡航速度， c 是单位时间燃料消耗量， $\frac{L}{D}$ 是升阻比， W_i, W_f 分别是飞行器的初始和最终质量。初始质量包括结构质量和燃料质量，而最终质量只包含结构质量。此外，还有另外一种目标函数的表达式：

$$I = \alpha C_D + \beta W \quad (1-2)$$

在设计过程中，最为重要的约束条件是结构的强度极限约束。考虑对于每个结构有限元，有

$$g_m = 1 - \frac{\sigma_m}{\sigma_y} \geq 0 \quad (1-3)$$

其中， σ_m 是第 m 个有限元元素的 von Mises 应力， σ_y 是材料的屈服极限。根据此，我们采用 KS 函数 (Kresisselmeier-Steinhauser function)：

$$KS = -\frac{1}{\rho} \ln \left[\sum_m e^{-\rho g_m} \right] \quad (1-4)$$

另外，优化的指标还可以是颤振速度、结构模态频率等动力学指标。

在气动弹性优化中，CFD 和有限元求解是计算量最大、成本最高的地方。因此，对于大规模优化问题，难点便在于在优化过程中尽可能少地调用气动弹性求解过程。优化算法通常可以分为两大类：无梯度算法与基于梯度算法^[3]。无梯度算法，例如遗传算法与粒子群算法，只需要获取函数在某个点的值，有很大概率获得全局最优解。然而，这类算法存在低收敛率、需要大量次数函数调用的问题，因此对于有大量设计变量的气动弹性优化问题并不适用。基于梯度的优化算法包括信

信赖域算法^[4]、有效集算法（包括序列二次规划算法^[5]）、内点法^[6]等方法。基于梯度的算法需要函数在某个点的值以及对应的梯度，具有收敛快、函数调用次数少等优点，适合于求解气动弹性优化问题。

梯度的求解方法分为多种，有一阶精度的前向差分法和后向差分法，二阶精度的中心差分法和复数差分法^[7]。前向差分法、后向差分法与中心差分法都属于有限差分法，存在计算效率不高、精度不高的问题。而复数差分法虽然精度高，但是计算成本也较大。此外，计算梯度也可以通过结合自动微分与伴随方法的方式^[8]，既可以显著减少运算量，又可以增加计算精度^[9]。伴随方法由于具有这样的优点，已经在气象学^[10]、计算经济学等领域受到大量关注，并在气动、结构优化问题上得到了应用。借助于梯度，可以给出目标变量关于设计变量的灵敏度，并可以用于求解优化问题。

1.3 本文的主要研究内容

本文对大展弦比飞机运用伴随方法优化性能指标进行研究，全文的安排如下：第二章介绍有限元基本方法，包括有限元的基本理论、载荷矩阵、刚度矩阵、有限元方程的求解等；第三章介绍气动力的计算方法，包括片条理论和涡格法；第四章介绍流固耦合的求解方法；第五章介绍伴随方法的基本思路，以及基于伴随方法对有限元方程和流固耦合方程的梯度计算；第六章是针对一段机翼进行气动弹性分析并运用伴随方法对机翼的性能指标进行优化，并满足强度约束；最后一章是总结与展望。

第 2 章 有限元方法

2.1 引言

有限元方法是一种用于求解微分方程组或积分方程组数值解的数值技术。这一解法基于完全消除微分方程，即将微分方程转化为代数方程组（稳定情形）；或将偏微分方程（组）改写为常微分方程（组）的逼近，这样可以用标准的数值技术（例如欧拉法，龙格—库塔法等）求解。在解偏微分方程的过程中，主要的难点是如何构造一个方程来逼近原本研究的方程，并且该过程还需要保持数值稳定性。目前有许多处理的方法，他们各有利弊。当区域改变时（就像一个边界可变的固体），当需要的精确度在整个区域上变化，或者当解缺少光滑性时，有限元方法是在复杂区域（像汽车、船体结构、输油管道）上解偏微分方程的一个很好的选择。例如，在正面碰撞仿真时，有可能在“重要”区域（例如汽车的前部）增加预先设定的精确度并在车辆的末尾减少精度（如此可以减少仿真所需消耗）；另一个例子是模拟地球的气候模式，预先设定陆地部分的精确度高于广阔海洋部分的精确度是非常重要的。

2.2 有限元分析的方法论

这里以泊松方程为例子，介绍有限元的理论和思想，泊松方程为：

$$\Delta u = f \quad (2-1)$$

在这里 Δ 代表的是拉普拉斯算子，而 f 和 φ 可以是在流形上的实数或复数值的方程。当流形属于欧几里得空间，而拉普拉斯算子通常表示为 ∇^2 ，因此泊松方程通常写成

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2} \right) \varphi(x, y, z) = f(x, y, z) \quad (2-2)$$

我们提出如下的带有边界条件的问题

$$\begin{aligned} -\Delta u &= f && \text{in } \Omega, \\ u &= 0 && \text{on } \partial\Omega. \end{aligned}$$

求解的区域为平面单位正方形， $\Omega = [0, 1]^2$ 。我们假设一个测试函数 φ ，并在上式的左边乘上这个函数，对整个区域 Ω 积分，可以得到

$$-\int_{\Omega} \varphi \Delta u = \int_{\Omega} \varphi f \quad (2-3)$$

由高斯散度定理（奥斯特罗格拉斯基公式）可以得到：

$$\int_{\Omega} \nabla \varphi \cdot \nabla u - \int_{\partial\Omega} \varphi \mathbf{n} \cdot \nabla u = \int_{\Omega} \varphi f \quad (2-4)$$

我们需要让 φ 也同样满足边界条件（用数学属于描述就是需要让测试函数 φ 来自方程解的切空间），因此上式可以写成

$$(\nabla \varphi, \nabla u) = (\varphi, f) \quad (2-5)$$

这里我们使用通用的内积符号 $(a, b) = \int_{\Omega} a b$ 。我们现在的思路就是找到一个解的近似 $u_h(\mathbf{x}) = \sum_j U_j \varphi_j(\mathbf{x})$ ，而 U_j 是我们要求的系数， φ_i 是我们用的形函数。因此可以得到

$$(\nabla \varphi_i, \nabla u_h) = (\varphi_i, f), \quad i = 0 \dots N-1. \quad (2-6)$$

现在需要求解一个矩阵代数方程

$$\mathbf{A}\mathbf{U} = \mathbf{F} \quad (2-7)$$

矩阵 \mathbf{A}, \mathbf{F} 定义为

$$A_{ij} = (\nabla \varphi_i, \nabla \varphi_j),$$

$$F_i = (\varphi_i, f).$$

显然，这样就有 $\sum_j \mathbf{A}_{ij} \mathbf{U}_j = \mathbf{F}_i$ 。前面看到我们是左乘 φ ，对于对称矩阵（自伴随的），问题是一样的，不过对于非对称的 \mathbf{A} ，那么经验表明这样会在计算中产生更大的误差。现在整理一下： $\mathbf{A}, \mathbf{U}, \mathbf{F}$ 都是稀疏矩阵，积分的计算采用高斯积分算法，这样就是全部思路。这样得到的拉普拉斯方程的解如图2-1所示。

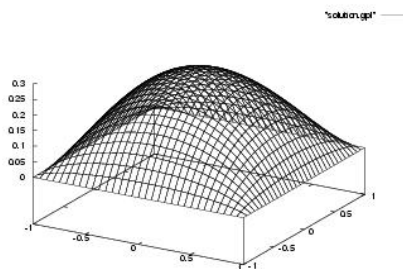


图 2-1 二维拉普拉斯方程的有限元解
Fig.2-1 FEM Solution on 2-D Laplace Equation

2.3 有限元求解弯曲单元模型

对于一段弯曲梁单元，可以设位移函数为

$$v(x) = f(v_1, v_2, \theta_1, \theta_2, x) \quad (2-8)$$

通过边界条件的方程可以得到

$$v(x) = \left(1 - \frac{2x^2}{L^2} + \frac{2x^3}{L^3}\right)v_1 + \left(x - \frac{2x^2}{L} - \frac{x^3}{L^2}\right)\theta_1 + \left(\frac{3x^2}{L^2} - \frac{2x^3}{L^3}\right)v_2 + \left(\frac{x^3}{L^2} - \frac{x^2}{L}\right)\theta_2 \quad (2-9)$$

也可以写成如下的形式：

$$v(x) = N_1(x)v_1 + N_2(x)\theta_1 + N_3(x)v_2 + N_4(x)\theta_2 \quad (2-10)$$

或者矩阵的形式

$$v(x) = \begin{bmatrix} N_1 & N_2 & N_3 & N_4 \end{bmatrix} \begin{bmatrix} v_1 \\ \theta_1 \\ v_2 \\ \theta_2 \end{bmatrix} \quad (2-11)$$

应力的计算可以表示为：

$$\sigma_x(x) = y_{\max} E \frac{d^2[N]}{dx^2} [\delta] \quad (2-12)$$

设载荷在结点上分别为 $[F_1 \quad M_1 \quad F_2 \quad M_2]^T$ ，由卡氏第一定理可以得到 \mathbf{F} 与 v, θ 之间的关系，从而得到刚度矩阵

$$[k_e] = \frac{EI_z}{L^3} \begin{bmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{bmatrix} \quad (2-13)$$

如果一段弯曲单元上受到集度为 q 的均布载荷，由虚功等效原理可以等效为结点上的载荷

$$\begin{bmatrix} F_{1q} \\ M_{1q} \\ F_{2q} \\ M_{2q} \end{bmatrix} = \begin{bmatrix} \frac{qL}{2} \\ \frac{qL^2}{12} \\ \frac{qL}{2} \\ \frac{-qL^2}{12} \end{bmatrix} \quad (2-14)$$

2.4 有限元求解具有弯曲和扭转的模型

若翼型为对称翼型，那么若力矩参考点在焦点（弦长 25% 处）时，没有俯仰力矩，焦点与压心重合。通常为了飞机的静稳定性，梁的位置不在焦点处，则梁会发生扭转。因此，单个具有弯曲和扭转的单元的刚度矩阵为（这里假定梁为圆柱形截面）

$$\begin{bmatrix} [k_{\text{bending}}]_{xy} & [0] \\ [0] & [k_{\text{torsion}}] \end{bmatrix} \begin{bmatrix} v_1 \\ \theta_{z1} \\ v_2 \\ \theta_{z2} \\ \theta_{x1} \\ \theta_{x2} \end{bmatrix} = \begin{bmatrix} f_{y1} \\ M_{z1} \\ f_{y2} \\ M_{z2} \\ M_{x1} \\ M_{x2} \end{bmatrix} \quad (2-15)$$

弯曲单元的子刚度矩阵与式 (2-13) 一致，而扭转子刚度矩阵为

$$[k_{\text{torsion}}] = \frac{JG}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \quad (2-16)$$

而扭转产生的切应力计算公式为：

$$\tau = G\gamma = G\rho \frac{d\varphi}{dx} = G\rho \frac{M}{GI_p} \quad (2-17)$$

圆柱形截面的极惯性矩为

$$I_p = \frac{\pi D^4}{32} \quad (2-18)$$

2.5 有限元方程的求解

有限元方程得到的矩阵通常是稀疏矩阵，通常用专门的稀疏矩阵算法对矩阵进行存储和计算。常见的适合用于解决这方面问题的程序库包括 Eigen 和 deal.II. 对于线性方程

$$Ku - f = 0 \quad (2-19)$$

当问题的自由度相对较小时,可以采用 LU 分解或者 Cholesky 分解的方式。Cholesky 分解在这类问题中得到广泛运用：对正定埃尔米特矩阵 A 进行 Cholesky 分解，即求矩阵 L 使得下式成立：

$$A = LL^* \quad (2-20)$$

式中， L 为一个下三角矩阵且所有对角元素均为正实数， L^* 为 L 的共轭转置矩阵。当 A 是正定矩阵时，Cholesky 分解是惟一的。Cholesky 分解的另一种形式——LDLT 分解的计算方式如下所示：

$$\begin{aligned} A = LDL^T &= \begin{pmatrix} 1 & 0 & 0 \\ L_{21} & 1 & 0 \\ L_{31} & L_{32} & 1 \end{pmatrix} \begin{pmatrix} D_1 & 0 & 0 \\ 0 & D_2 & 0 \\ 0 & 0 & D_3 \end{pmatrix} \begin{pmatrix} 1 & L_{21} & L_{31} \\ 0 & 1 & L_{32} \\ 0 & 0 & 1 \end{pmatrix} \\ &= \begin{pmatrix} D_1 & & \\ L_{21}D_1 & L_{21}^2D_1 + D_2 & \\ L_{31}D_1 & L_{31}L_{21}D_1 + L_{32}D_2 & L_{31}^2D_1 + L_{32}^2D_2 + D_3 \end{pmatrix} \quad \text{(symmetric)} \end{aligned} \quad (2-21)$$

常见的求解线性方程组的算法及其对应的特点以表格的形式罗列在2-1：对于有更大自由度的有限元模型（超过 10^5 ），通常会采用一些别的求解方式。例如，对于稀疏矩阵的线性方程组求解，一般采用迭代的算法进行求解。对于自伴随矩阵，可以采用共轭梯度下降法求解，而对于一般的方阵则采用 BiCGStab 算法求解。对于大规模稀疏线性方程组，迭代算法能够比矩阵分解算法更高效。但是，由于 LDLT 分解也存在一定优势，国内黄江涛^[11]所编写的气动弹性求解器亦是用 LDLT 分解求解线性方程组。

这样，对于一段梁受到载荷作用，根据弯曲单元写出装配矩阵，并通过上面的

表 2-1 各种求解线性方程组算法的特点

Table2-1 comparison between different matrix decomposition Algorithms

算法	对矩阵的要求	速度（对小 到中等规模 矩阵）	速度（对大 规模矩阵）	精度
PartialPivLU	不可逆	++	++	+
FullPivLU	无	-	--	+++
Householder QR	无	++	++	+
行旋转 QR	无	+	-	+++
全旋转 QR	无	-	--	+++
完全正交分解	无	+	-	+++
LL^T	正定	+++	+++	+
LDL^T	正定或半负定	+++	+	++
BDCSVD	无	-	-	+++
JacobiSVD	无	-	---	+++

方法求解式 (2-19)，可以得到梁的位移、应力等信息。编写梁单元的有限元程序，线性方程组的求解算法为 Cholesky 分解中的 LDLT 分解，求解有限元方程并得到的形变图如图2-2所示。

同样，根据基尔霍夫平板理论，列出板单元的刚度矩阵，也可以编写板单元的有限元程序，得到的形变图如2-3所示。



图 2-2 梁的有限元求解
Fig.2-2 Beam's FEM solution

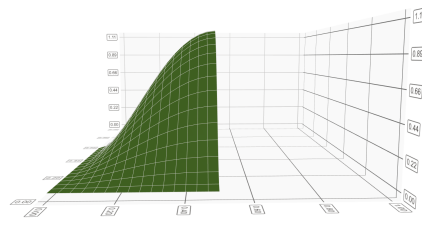


图 2-3 板的有限元求解
Fig.2-3 Plate's FEM solution

有限元方法还可以求解超静定梁的问题，如图2-4所示。



图 2-4 超静定梁的有限元求解
Fig.2-4 Statically Indeterminate Beam's FEM solution

2.6 小结

本章对基本的有限元理论进行研究，并具体研究了弯曲单元、扭转单元、板单元的刚度矩阵并编写求解器。通过求解器求解常见的力学问题，可以看出有限元方法具有精度高、计算速度快、能够处理不同种类力学问题等优点，将为后面机翼的气动弹性分析中的结构分析打下基础。

第 3 章 气动计算方法

3.1 引言

对于航空器的气动力计算，通常有片条理论、升力线方法、涡格法、CFD 方法等。片条理论最为简单，但精确度较低；CFD 方法通常最为精确，但求解的成本较大。前面三种方法通常不考虑气体的粘性，或者在计算后对粘性进行修正，因此不适合雷诺数较小的流动状况。在考虑气动弹性的问题时，通常使用片条理论、升力线方法、涡格法而较少使用 CFD 方法，以寻求在计算效率和精确度之间的平衡。

3.2 片条理论

片条理论是一种忽略了下洗的计算方法，能够简单快捷地计算升力分布。在机翼的展弦比大于 5 的时候，能够给出足够精确的结果。将机翼分成若干个片条，各个片条的升力为

$$L_i = C_L^\alpha \alpha_i S_i \quad (3-1)$$

为了修正三元流动的影响，对于展向升力分布接近椭圆分布的直机翼，其升力线斜率可作如下修正^[12]：

$$\frac{\partial C_L}{\partial \alpha} = \left(\frac{\partial C_L}{\partial \alpha} \right)_\infty \frac{\lambda}{\lambda + 2} \quad (3-2)$$

式中 λ ——机翼展弦比；

$\left(\frac{\partial C_L}{\partial \alpha} \right)_\infty$ ——二维翼型的升力线斜率。

在处理静气动弹性问题时，由于翼面的扭转变形，攻角沿翼展变化，这时可以用另一个修正公式^[13]，即

$$\frac{\partial C_L}{\partial \alpha} = \left(\frac{\partial C_L}{\partial \alpha} \right)_\infty \frac{\lambda}{\lambda + 4} \quad (3-3)$$

通常处理气动力计算时，会引入气动力影响矩阵 \mathbf{A} ，片条理论的气动力影响矩阵为

$$\mathbf{A} = \text{diag} \left[\frac{1}{C_L^\alpha c_i} \quad \cdots \quad \frac{1}{C_L^\alpha c_n} \right] \quad (3-4)$$

式中， C_L^α 应按风洞试验结果或按理论公式计算， c 为弦长。那么攻角与升力系数满足以下关系

$$As = \alpha \quad (3-5)$$

式中 s ——各点处的 cC_L 组成的列阵, $s = [c_1 C_L(y_1) \cdots c_n C_L(y_n)]^T$;
 α ——各点处的攻角阵列。

3.3 升力线法

由于片条理论忽略了下洗, 因此会导致气动力计算的不准确, 并且无法计算诱导阻力。经典的升力线方法考虑翼型的升力系数随攻角线性变化, 并且不考虑气体的粘性。由比奥-萨伐尔定律, 一段强度为 Γ 涡丝诱导出来的诱导速度的表达式为

$$dV = \frac{\Gamma}{4\pi} \frac{dl \times r}{|r|^3} \quad (3-6)$$

对于机翼上任意一点 y_0 , 机翼所有地方对这里总的诱导速度的表达式可以写为

$$w(y_0) = -\frac{1}{4\pi} \int_{-b/2}^{b/2} \frac{(d\Gamma/dy)dy}{y_0 - y} \quad (3-7)$$

那么这里的诱导攻角为

$$\alpha_i(y_0) = \arctan\left(\frac{-w(y_0)}{V_\infty}\right) = \frac{1}{4\pi V_\infty} \int_{-b/2}^{b/2} \frac{(d\Gamma/dy)dy}{y_0 - y} \quad (3-8)$$

由茹科夫斯基升力公式, 在 y_0 处的升力表达式

$$L' = \frac{1}{2} \rho_\infty V_\infty^2 c(y_0) c_l = \rho_\infty V_\infty \Gamma(y_0) \quad (3-9)$$

且有

$$\alpha_{\text{eff}} = \alpha - \alpha_i \quad (3-10)$$

因此可以得到普朗特升力线理论的基本方程

$$\alpha(y_0) = \frac{\Gamma(y_0)}{\pi V_\infty c(y_0)} + \alpha_0 + \frac{1}{4\pi} \int_{-b/2}^{b/2} \frac{(d\Gamma/dy)dy}{y_0 - y} \quad (3-11)$$

而诱导阻力的公式为

$$D'_i = L'_i \sin \alpha_i \approx L'_i \alpha_i \quad (3-12)$$

$$D_i = \rho_\infty V_\infty \int_{-b/2}^{b/2} \Gamma(y) \alpha_i(y) dy \quad (3-13)$$

通过对上式进行离散化处理, 采用数值积分的方式, 并进行迭代计算环量 $\Gamma(y)$ 的

分布^[14]，可以在计算机上对方程（3-11）进行数值求解。

3.4 涡格法

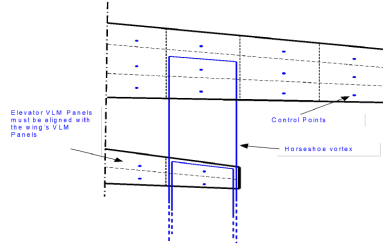


图 3-1 涡格法示意图

Fig. 3-1 Sketch of Vortex Panel Method

如图3-1所示，由于针对 CFD 求解机翼流场的成本巨大，因此工程上常常采用涡格法求解机翼流场。涡格法在面元上布置点源、点涡或者偶极子，通过控制点法向速度为 0 的边界条件，可以比较精确得解出下洗、诱导阻力、升力等。在势流理论中，的一个基本解是源（汇）。当这个元素的强度是正的时候，就被称为源，负的称为汇。在 (x_0, y_0, z_0) 的源作用在位于点 $P(x, y, z)$ 的点的速度势为

$$\begin{aligned}\phi(P(x, y, z)) &= -\frac{\sigma}{4\pi|\mathbf{r} - \mathbf{r}_0|} && \text{(球坐标)} \\ &= \frac{-\sigma}{4\pi\sqrt{(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2}} && \text{(直角坐标)}\end{aligned}\quad (3-14)$$

可以将基本解分布在表面上，通常称为板块或者面元。因此分布在某个面上的源作用在点 $P(x, y, z)$ 的速度势能够通过积分式 (3-14) 得到

$$\phi(P(x, y, z)) = \frac{-1}{4\pi} \int_S \frac{-\sigma(x_0, y_0, z_0)}{(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2} dS \quad (3-15)$$

而若在表面布置偶极子或者点涡，则可以得到环量并计算升力。由涡格法可以得到方程：

$$\mathbf{A}\mathbf{\Gamma} = \mathbf{v} \quad (3-16)$$

其中 \mathbf{A} 是气动力影响系数矩阵， $\mathbf{\Gamma}$ 是各个面元的环量组成的向量， \mathbf{v} 是边界条件向量，也就是各个面元的当地攻角。

编写的涡格法程序，得到某个未优化的机翼升力分布和弯矩分布如图3-2,3-3所示。

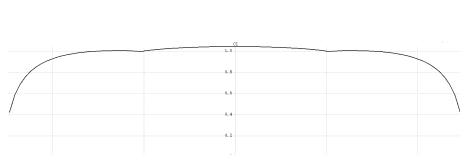


图 3-2 涡格法得到的机翼升力分布

Fig.3-2 The Lift Distribution of a Wing Calculated by VPM

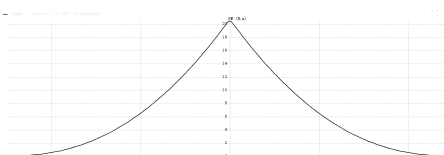


图 3-3 涡格法得到的机翼弯矩图

Fig. 3-3 The Bending Moment Distribution of a Wing Calculated by VPM

涡格法由于考虑了下洗，因此也能够计算出机翼的诱导阻力，诱导阻力分布如图3-4所示。

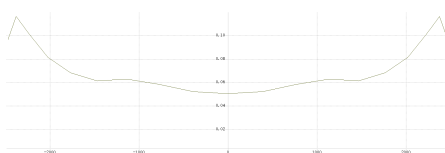


图 3-4 涡格法得到的机翼诱导阻力分布

Fig.3-4 The Induced Drag Distribution of a Wing Calculated by VPM

3.5 小结

本章介绍了工程中常用的计算气动力的方法：片条理论，升力线方法，涡格法和 CFD 方法，并编写相应的气动力求解器，为流固耦合分析中的气动力部分进行铺垫。

第 4 章 流固耦合方程的求解

4.1 引言

“气动弹性力学”概念最早由航空工程师在 20 世纪 30 年代引出，到现在已经发展成为了一门独立的力学学科，其主要考虑气动力与弹性体之间的相互作用，进而研究弹性结构在气流中的力学行为。气流中的弹性结构在空气动力的作用下回发生弹性变形或振动，而结构变形又会让气动力重新分布，这种结构与气动间的相互耦合作用使弹性体产生各类气动弹性现象。为了明确气动弹性力学的分类以及了解气动弹性问题的研究内容，英国学者 Collar 在 1946 年绘制了气动弹性力学三角形，生动具体地表示了气动弹性力学所涉及的各个学科以及各学科间的相互联系，如图4-1所示。

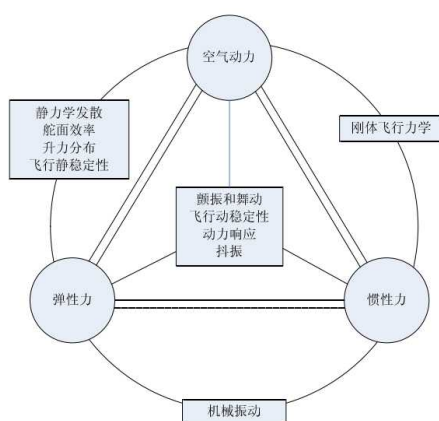


图 4-1 气动弹性力学三角形

Fig. 4-1 Aeroelasticity Triangle

4.2 流固耦合的残差方程

流体力学残差方程为

$$R = A w - \alpha = 0 \quad (4-1)$$

有限元残差方程为

$$S = K u - f = 0 \quad (4-2)$$

气动弹性力学的求解器通过耦合结构的有限元方程和气动的动力学方程，在收敛求解的过程中，通过分别求解结构残差方程式 (4-1) 和气动残差方程 (4-2)，并互相使用求解出的气动力和位移数据，不断迭代，直到收敛。在这里我们仅仅考虑静气动弹性问题，因此忽略动力学相关的如模态频率，时间步长等因素。

如图4-2所示，当流体力学的求解器第一次被调用时，结构的位移被初始化为0。当流体力学求解器进行了一次求解以后，机翼气动面上的动压被转化为了节点上的力以及力矩，然后调用有限元求解器求解位移，新的位移用来重新求解在机翼上产生的气动力。这个过程一直持续直到气动力（流场的流动状况）和结构的位移收敛，也就是当残差方程 (4-1) 和 (4-2) 的范数小于某个很小的常数 ϵ 时迭代停止，或者相对误差满足方程组4-3时迭代停止。

$$\begin{aligned} \|R(\mathbf{w}^{(n)}, \mathbf{u}^{(n)})\|_2 &< \epsilon_r \|R(\mathbf{w}^{(0)}, \mathbf{u}^{(0)})\| \\ \|S(\mathbf{w}^{(n)}, \mathbf{u}^{(n)})\|_2 &< \epsilon_r \|S(\mathbf{w}^{(0)}, \mathbf{u}^{(0)})\| \end{aligned} \quad (4-3)$$

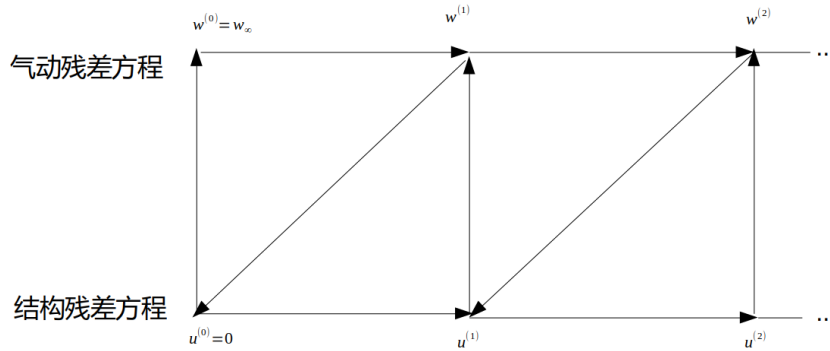


图 4-2 气动弹性迭代求解示意图

Fig. 4-2 Schematic Representation of the Aero-Structural Iteration Procedure

在气动弹性耦合方程的状态求解过程中，在这里我们由于使用的都是线性的模型，因此求解的时间成本消耗主要在矩阵的分解上。每迭代一步，无论是气动力影响系数矩阵还是有限元刚度矩阵都会改变，因此求解线性方程组需要对矩阵重新做分解。因此，对于设计变量较多，方程的自由度数较大时，如果需要大量调用此求解过程，时间成本会非常高。对此，文献^[15]提出了改进的，支持并行化的，分块的求解方法，包括非线性分块高斯-赛德尔迭代，非线性分块雅各比迭代，牛顿-科里洛夫迭代和近似牛顿-科里洛夫迭代。

4.3 非线性分块高斯-赛德尔

非线性高斯赛德尔迭代的步骤是在每一步迭代更新时去求解一系列的，线性化的两个残差方程。步骤可以写成如下的式子：

$$\frac{\partial \mathbf{R}}{\partial \mathbf{w}} \Delta \mathbf{w}^{(n)} = -\mathbf{A}(\mathbf{w}^{(n)}, \mathbf{u}^{(n)}) \quad (4-4)$$

$$\mathbf{K}^{(n)} \Delta \mathbf{u}^{(n)} = -\mathbf{S}(\mathbf{w}^{(n+1)}, \mathbf{u}^{(n)}) \quad (4-5)$$

可以看出， $\mathbf{w}^{(n+1)} = \mathbf{w}^{(n)} + \alpha_R^{(n)} \Delta \mathbf{w}^{(n)}$ 会先于 $\mathbf{u}^{n+1} = \mathbf{u}^{(n)} + \alpha_S^{(n)} \Delta \mathbf{u}^{(n)}$ 得到更新。参数 $\alpha_R^{(n)}$ 和 $\alpha_S^{(n)}$ 是需要调整的迭代求解的阻尼因子。直接按照很小的误差求解上式不会使得这个算法在计算上的高效。因此，我们需要下面的这个停止条件：

$$\|\mathbf{R}(\mathbf{w}^{(n)}, \mathbf{u}^{(n)}) + \frac{\partial \mathbf{R}}{\partial \mathbf{w}} \Delta \mathbf{w}^{(n)}\|_2 < \epsilon_{gs} \|\mathbf{R}(\mathbf{w}^{(n)}, \mathbf{u}^{(n)})\|_2 \quad (4-6)$$

通常将 ϵ 设置为 10^{-3} 。而阻尼参数的选择对于这个算法的鲁棒性和效率也十分关键。在优化问题中，结构和气动状态变量的变化范围或许会非常大，因此必须考虑到一个足够宽的分析范围。可以选择 $\alpha_R^{(n)} = 1$ 并动态选择 $\alpha_S^{(n)}$ ：

$$\begin{aligned} \tilde{\alpha}_S &= \alpha_S^{(n-1)} \left(1 - \frac{(\Delta \mathbf{u}^{(n)} - \Delta \mathbf{u}^{(n-1)})^T \Delta \mathbf{u}^{(n)}}{\|\Delta \mathbf{u}^{(n)} - \Delta \mathbf{u}^{(n-1)}\|_2} \right) \\ \alpha_S &= \max(\alpha_{\text{low}}, \min(1, \tilde{\alpha}_S)) \end{aligned} \quad (4-7)$$

通常选择 $\alpha_{\text{low}} = 0.25, \alpha_S^{(0)} = 0.5$ 。

4.4 非线性分块雅各比

一个非线性分块高斯赛德尔迭代的替代方法是求解完两个方程后对两组状态变量同时进行更新，这就是非线性分块雅各比迭代：

$$\frac{\partial \mathbf{R}}{\partial \mathbf{w}} \Delta \mathbf{w}^{(n)} = -\mathbf{A}(\mathbf{w}^{(n)}, \mathbf{u}^{(n)}) \quad (4-8)$$

$$\mathbf{K}^{(n)} \Delta \mathbf{u}^{(n)} = -\mathbf{S}(\mathbf{w}^{(n)}, \mathbf{u}^{(n)}) \quad (4-9)$$

然后同时更新 $\mathbf{w}^{(n+1)} = \mathbf{w}^{(n)} + \alpha_R^{(n)} \Delta \mathbf{w}^{(n)}$ 和 $\mathbf{u}^{n+1} = \mathbf{u}^{(n)} + \alpha_S^{(n)} \Delta \mathbf{u}^{(n)}$ 。这样处理的好处在于可以将两组方程并行求解，增加方程求解的并行度以减小求解成本。

4.5 牛顿-科里洛夫方法

牛顿-科里洛夫方法是将整个残差方程进行线性化。记整个残差方程为

$$\mathbf{T} = \begin{bmatrix} \mathbf{R}(\mathbf{w}, \mathbf{u}) \\ \mathbf{S}(\mathbf{w}, \mathbf{u}) \end{bmatrix} = \mathbf{0} \quad (4-10)$$

并记所有状态变量为一个列向量, $\mathbf{q}^T = [\mathbf{w}^T, \mathbf{u}^T]$, 则牛顿-科里洛夫方法使用下式更新状态向量 $\mathbf{q}^{(n)}$:

$$\frac{\partial \mathbf{T}}{\partial \mathbf{q}} \Delta \mathbf{q}^{(n)} = -\mathbf{T}(\mathbf{q}^{(n)}) \quad (4-11)$$

如果迭代的起始点距离真实解足够近, 且雅各比矩阵非奇异的话, 那么牛顿法将以平方速率收敛。为了让迭代收敛, 牛顿法需要一些全局的搜索策略来确保迭代点在向真实解靠近。常见的全局搜索策略包括伪瞬态延续法。文献^[15]指出对于气动弹性问题, 全局搜索策略是不必要的。

对于每一步更新, 近似求解式 (4-11) 比精确求解效率更高, 只需满足误差 $\epsilon_{nk} = 10^{-3}$

$$\|\mathbf{T}(\mathbf{q}^{(n)}) + \frac{\partial \mathbf{T}}{\partial \mathbf{q}} \Delta \mathbf{q}^{(n)}\|_2 < \epsilon_{nk} \|\mathbf{T}(\mathbf{q}^{(n)})\|_2 \quad (4-12)$$

并更新 $\mathbf{q}^{(n+1)} = \mathbf{q}^{(n)} + \Delta \mathbf{q}^{(n)}$. 为了保证式 (4-3) 得到满足, 牛顿-科里洛夫方法的停止条件为

$$\|\mathbf{T}(\mathbf{q}^{(n)})\|_2 < \epsilon_r \min(\|\mathbf{R}(\mathbf{q}^{(0)})\|_2, \|\mathbf{S}(\mathbf{q}^{(0)})\|_2) \quad (4-13)$$

4.6 近似牛顿-科里洛夫方法

牛顿-科里洛夫方法的迭代效率为平方收敛效率, 理论上会很快收敛, 但是由于 $\frac{\partial \mathbf{T}}{\partial \mathbf{q}}$ 的非对角项块 $\frac{\partial \mathbf{R}}{\partial \mathbf{u}}, \frac{\partial \mathbf{S}}{\partial \mathbf{w}}$, 这两个耦合项的导数的计算无论采用何种方式都会花费大量时间, 因此牛顿-科里洛夫方法明显慢于非线性分块高斯-赛德尔迭代。为了提高效率, 通过一些方式简单地直接估算耦合项的导数, 而其它实现细节与牛顿-科里洛夫方法相同, 这就是近似牛顿-科里洛夫方法。

4.7 静气动弹性机翼升力重新分布求解实例

设某段矩形机翼翼展为 5 米, 展弦比为 9, 机翼梁的为圆柱形实心梁, 梁的直径为 6cm, 梁的材料的弹性模量为 2GPa, 泊松比为 0.2, 梁位于弦长的 30% 处。假设空气密度为 $\rho = 1.225\text{kg/m}^3$, 风速为 18m/s, 机翼对气流的攻角为 5° , 机翼的升力线斜率为 5/rad. 将半段机翼的主梁考虑为具有弯曲和扭转特性的悬臂梁列出有限元方程, 并根据片条理论列出气动力方程, 总共划分为 10 个网格, 按照非线性高

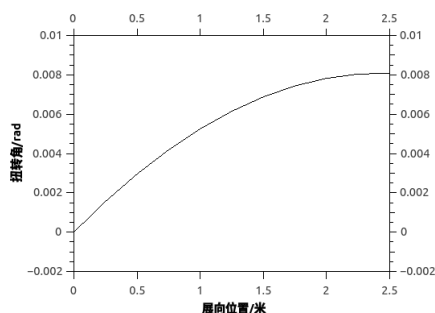


图 4-3 弹性机翼的扭转角

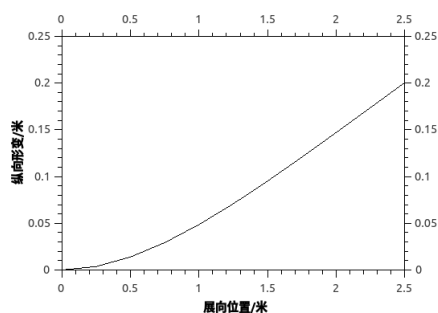


图 4-4 弹性机翼的纵向形变

Fig.4-3 Deflection Angle of an Elastic Wing Fig. 4-4 Displacements of an Elastic Wing

斯-赛德尔迭代求解机翼在考虑气动弹性作用下的升力分布，经过9次迭代，最终得到弹性机翼的升力为 $L = 128.306\text{N}$ ，而考虑为刚体的机翼的升力为 $L_0 = 120.264\text{N}$ ，可见由于刚心位于焦点之后，升力的作用产生的扭转角导致翼型对气流的攻角增大，因此产生的总升力大于刚性机翼产生的总升力，在此处给定的来流下升力增大了6.7%。弹性机翼沿着展向的扭转角变化和纵向形变如图4-3和图4-4所示。

4.8 小结

本章介绍了几种求解流固耦合方程的不同的迭代方法，并结合片条理论与有限元方法，对一段实际的半机翼在气流中的状况进行气动弹性分析，运用非线性分块高斯-赛德尔迭代，迭代十步左右即收敛，得到的结果是弹性机翼由于刚心在焦点之后由于扭转变形会导致对气流的攻角增大，因此弹性机翼会产生更大的升力。对于此问题，还应该进一步进行试验验证，以分析计算结果的误差。

第 5 章 伴随方法

5.1 引言

由于基于梯度的优化算法需要利用性能指标关于设计变量的导数，因此如何高效而精确地计算出导数是一个关键问题。在本章介绍几种常用的求解梯度的算法并对比各自的优缺点，并重点介绍伴随方法的对应理论。

5.2 梯度的计算方法

5.2.1 有限差分法

这个的原理是利用泰勒级数的近似表达式：

$$\frac{\partial y}{\partial x} = \frac{y|_{x+\delta} - y|_x}{\delta} \quad (5-1)$$

上式为后向差分的表达式，为一阶精度。有限差分法易于理解，在编程上也易于实现，因此有大量的应用，例如流体力学的一种常见方法就是有限差分法。为了提高精度，通常会采用中心差分法。然而，有限差分法存在的问题包括数值稳定性低、精度不高、计算效率不高等，因此不适合一些特殊的场合的梯度计算。如果用中心差分法替代前向差分法或后向差分法，那么精度能够有所提高，但是计算耗时也会增加。

5.2.2 复数差分法

考虑一个解析函数 $F(z)$ ，这意味着该函数无限可微且能够在光滑的延拓的复平面上，那么对于实数 x_0 和 h ，根据泰勒级数有

$$F(x_0 + ih) = F(x_0) + ihF'(x_0) - \frac{h^2 F''(x_0)}{2!} \dots \quad (5-2)$$

取虚部可以得到

$$F(x_0 + ih) = \frac{\text{Im}(F(x_0 + ih))}{h} + O(h^2) \quad (5-3)$$

因此得到

$$F'(x_0) = \frac{\text{Im}(F(x_0 + ih))}{h} \quad (5-4)$$

复数差分法由于精度为二阶精度，因此拥有更高的数值稳定性以及计算精度^[7]，而

且编程也并不复杂，但是计算效率甚至低于有限差分法。

5.2.3 自动微分法

在伴随方法中，仍然需要求得函数对于某个变量的导数或偏导数，通常采用自动微分的算法。自动微分是算导数的最优方法，比符号计算、有限微分更快更精确。自动微分已经广泛应用在优化领域，包括人工神经网络的训练算法 back-propagation。要解连续优化或非线性方程，自动微分是不二的选择。自动微分的原理即为在函数的每一次运算中，利用链导法则对每一次运算进行微分。自动微分将符号微分法应用于最基本的算子，比如常数，幂函数，指数函数，对数函数，三角函数等，然后代入数值，保留中间结果，最后再应用于整个函数。因此它应用相当灵活，可以做到完全向用户隐藏微分求解过程，由于它只对基本函数或常数运用符号微分法则，所以它可以灵活结合编程语言的循环结构，条件结构等。自动微分可以分为正序模式与倒序模式。正序模式中导数沿着代码的执行路径传播，倒序模式则是要求从输出到输入节点，按照先后依赖关系，对各个节点求取输出对于当前节点的梯度，一旦搜集好对应输出边节点关于当前节点导数，那么当前节点的导数便可以由链导法则计算得出。通常使用的方式是倒序模式，因为许多微分过程需要倒序模式才能高效求解。在 GitHub 上有许多关于自动微分的优秀 C++ 实现，如 ceres 中的 Jet.h, codipack, adept 等。

5.3 伴随方法与机翼优化

给定某个依赖于设计参数 \mathbf{p} 离散偏微分方程或者别的某些方程的解 \mathbf{x} ，我们常常会希望计算某个关于基于设计变量以及解的函数 $g(\mathbf{x}, \mathbf{p})$ 。我们也需要求得 g 关于 \mathbf{p} 的梯度。

5.3.0.1 线性方程组的伴随方法

假设列向量 \mathbf{x} 是线性方程组 $\mathbf{Ax} = \mathbf{b}$ 的解，其中 \mathbf{A} 是关于设计变量 \mathbf{p} 的 $M \times M$ 的实数矩阵， \mathbf{b} 是关于设计变量 \mathbf{p} 的实数列向量。为了直接得到 g 的梯度，有

$$\frac{dg}{d\mathbf{p}} = \frac{\partial g}{\partial \mathbf{p}} + \frac{\partial g}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial \mathbf{p}} \quad (5-5)$$

由于 g 是一个给定的函数，假设 $\frac{\partial g}{\partial \mathbf{p}}, \frac{\partial g}{\partial \mathbf{x}}$ 不难求得。另一方面， $\frac{\partial \mathbf{x}}{\partial \mathbf{p}}$ 的计算会很困难，因为对于每一个设计变量 p_i ，需要计算 $\frac{\partial \mathbf{x}}{\partial p_i} = \mathbf{A}^{-1}(\frac{\partial \mathbf{b}}{\partial p_i} - \frac{\partial \mathbf{A}}{\partial p_i} \mathbf{x})$ 。也就是说，我们需要求解 p 次 M 阶矩阵线性方程，这对于 p, m 特别大的时候是不现实的。

我们可以将问题这样解决：

$$\frac{\partial g}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial \mathbf{p}} = \frac{\partial g}{\partial \mathbf{x}} [A^{-1}(\frac{\partial \mathbf{b}}{\partial \mathbf{p}} - \frac{\partial A}{\partial \mathbf{p}} \mathbf{x})] = [\frac{\partial g}{\partial \mathbf{x}} A^{-1}](\frac{\partial \mathbf{b}}{\partial \mathbf{p}} - \frac{\partial A}{\partial \mathbf{p}} \mathbf{x}) \quad (5-6)$$

并令

$$A^T \lambda = \frac{\partial g}{\partial \mathbf{x}}^T \quad (5-7)$$

则可以得到：

$$\frac{dg}{d\mathbf{p}} = \frac{\partial g}{\partial \mathbf{p}} - \lambda^T (\frac{\partial \mathbf{b}}{\partial \mathbf{p}} - \frac{\partial A}{\partial \mathbf{p}} \mathbf{x}) \quad (5-8)$$

通过伴随方法，这样可以只求解一次 p 次 M 阶矩阵线性方程式（5-7）。在这里 $\frac{\partial A}{\partial \mathbf{p}}$ 是三阶张量。

5.3.0.2 非线性方程组的伴随方法

如果 \mathbf{x} 满足更一般的、非线性的方程组 $\mathbf{f}(\mathbf{x}, \mathbf{p}) = \mathbf{0}$ ，伴随方法的求解过程与线性方程几乎一致。对函数 f 求微分，得到 $\frac{\partial f}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial \mathbf{p}} + \frac{\partial f}{\partial \mathbf{p}} = \mathbf{0}$ ，因此 $\frac{\partial \mathbf{x}}{\partial \mathbf{p}} = -\frac{\partial f}{\partial \mathbf{x}}^{-1} \frac{\partial f}{\partial \mathbf{p}}$ ，因此得到

$$\frac{dg}{d\mathbf{p}} = \frac{\partial g}{\partial \mathbf{p}} + \frac{\partial g}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial \mathbf{p}} = \frac{\partial g}{\partial \mathbf{p}} - \frac{\partial g}{\partial \mathbf{x}} [\frac{\partial f}{\partial \mathbf{x}}^{-1} \frac{\partial f}{\partial \mathbf{p}}] = \frac{\partial g}{\partial \mathbf{p}} - [\frac{\partial g}{\partial \mathbf{x}} \frac{\partial f}{\partial \mathbf{x}}^{-1}] \frac{\partial f}{\partial \mathbf{p}} \quad (5-9)$$

同样，令

$$\frac{\partial \mathbf{f}}{\partial \mathbf{x}}^T \lambda = \frac{\partial g}{\partial \mathbf{x}}^T \quad (5-10)$$

则最终可以得到：

$$\frac{dg}{d\mathbf{p}} = \frac{\partial g}{\partial \mathbf{p}} - \lambda^T \frac{\partial \mathbf{f}}{\partial \mathbf{p}} \quad (5-11)$$

5.4 伴随方法与有限元模型的结合

5.4.1 简化模型

有限元的基本方程可以写为 (2-19)，应力的计算可以表示为：

$$\sigma = \mathbf{S} \mathbf{u} \quad (5-12)$$

由于在这里仅仅考虑静载荷问题，因此有 $\frac{\partial \mathbf{S}}{\partial \mathbf{u}} = 0$, $\frac{\partial \mathbf{f}}{\partial \mathbf{p}} = 0$, $\frac{\partial \mathbf{f}}{\partial \mathbf{u}} = 0$ ，因此有

$$\mathbf{K} \frac{\partial \mathbf{u}}{\partial \mathbf{p}} + \frac{\partial \mathbf{K}}{\partial \mathbf{p}} \mathbf{u} = \mathbf{0} \quad (5-13)$$

$$\frac{\partial \sigma}{\partial \mathbf{p}} = \mathbf{S} \frac{\partial \mathbf{u}}{\partial \mathbf{p}} \quad (5-14)$$

可以得到

$$\frac{\partial \sigma}{\partial \mathbf{p}} = \mathbf{S} \mathbf{K}^{-1} \frac{\partial \mathbf{K}}{\partial \mathbf{p}} \mathbf{u} \quad (5-15)$$

这个方程的伴随方程是

$$\mathbf{K} \boldsymbol{\lambda} = \mathbf{S} \quad (5-16)$$

$\boldsymbol{\lambda}^T$ 即为伴随向量。因此有

$$\frac{\partial \sigma}{\partial \mathbf{p}} = \boldsymbol{\lambda}^T \frac{\partial \mathbf{K}}{\partial \mathbf{p}} \mathbf{u} \quad (5-17)$$

对于 $\mathbf{A}_{m \times n} \mathbf{B}_{n \times p}$ 的矩阵乘法运算，其计算复杂度为 $O(mnp)$ 。因此例如对于上述问题，若 \mathbf{p} 设计变量的数目很大，而 \mathbf{S}, \mathbf{K} 的阶数相对较小时，先计算伴随向量 $\mathbf{S} \mathbf{K}^{-1}$ ，再乘矩阵 $\frac{\partial \mathbf{K}}{\partial \mathbf{p}} \mathbf{u}$ ，运算量将小于计算 $\mathbf{S}(\mathbf{K}^{-1} \frac{\partial \mathbf{u}}{\partial \mathbf{p}} \mathbf{u})$

5.4.2 复杂模型

有限元的基本方程可以写为式 (4-1) 的形式，由于在这里仅仅考虑静载荷问题，因此有 $\frac{\partial f}{\partial \mathbf{p}} = 0, \frac{\partial f}{\partial \mathbf{u}} = 0$ ，因此有

$$\mathbf{K} \delta \mathbf{u} = -\frac{\partial \mathbf{K}}{\partial \mathbf{p}} \mathbf{u} \delta \mathbf{p} \quad (5-18)$$

如果 g 代表一个性能指标（如应力），有

$$\delta g = \frac{\partial g}{\partial \mathbf{p}} \delta \mathbf{p} + \frac{\partial g}{\partial \mathbf{u}} \delta \mathbf{u} \quad (5-19)$$

记 I 为性能指标，约束方程为

$$R(\mathbf{u}, \mathbf{f}, \mathbf{p}) = \mathbf{K} \mathbf{u} - \mathbf{f} = 0 \quad (5-20)$$

$$\delta I = \frac{\partial I}{\partial \mathbf{u}} \delta \mathbf{u} + \frac{\partial I}{\partial \mathbf{f}} \delta \mathbf{f} + \frac{\partial I}{\partial \mathbf{p}} \delta \mathbf{p} \quad (5-21)$$

$$\delta R = \frac{\partial R}{\partial \mathbf{u}} \delta \mathbf{u} + \frac{\partial R}{\partial \mathbf{f}} \delta \mathbf{f} + \frac{\partial R}{\partial \mathbf{p}} \delta \mathbf{p} \quad (5-22)$$

这就可以看做是一个带约束的优化问题。带约束的优化问题通常可以使用拉格朗日乘数法则。设伴随向量为 $\boldsymbol{\psi}$ ，也可以看作拉格朗日乘子。用拉格朗日乘数法则可以得到

$$\delta I = \left[\frac{\partial I}{\partial \mathbf{u}} - \boldsymbol{\psi}^T \frac{\partial R}{\partial \mathbf{u}} \right] \delta \mathbf{u} + \left[\frac{\partial I}{\partial \mathbf{f}} - \boldsymbol{\psi}^T \frac{\partial R}{\partial \mathbf{f}} \right] \delta \mathbf{f} + \left[\frac{\partial I}{\partial \mathbf{p}} - \boldsymbol{\psi}^T \frac{\partial R}{\partial \mathbf{p}} \right] \delta \mathbf{p} \quad (5-23)$$

选择伴随向量 ψ 满足

$$\left(\frac{\partial \mathbf{R}}{\partial \mathbf{u}}\right)^T \psi = \frac{\partial I}{\partial \mathbf{u}} \quad (5-24)$$

且记

$$\mathbf{G}_f = \frac{\partial I^T}{\partial \mathbf{f}} - \psi^T \frac{\partial \mathbf{R}}{\partial \mathbf{f}}$$

$$\mathbf{G}_p = \frac{\partial I^T}{\partial \mathbf{p}} - \psi^T \frac{\partial \mathbf{R}}{\partial \mathbf{p}}$$

则有

$$\delta I = \mathbf{G}_f \delta \mathbf{f} + \mathbf{G}_p \delta \mathbf{p} \quad (5-25)$$

比如 $\mathbf{R} = \mathbf{K}\mathbf{u} - \mathbf{f}$ ，有 $\frac{\partial \mathbf{R}}{\partial \mathbf{u}} = \mathbf{K}$ ，可得

$$\mathbf{K}^T \psi = \frac{\partial I}{\partial \mathbf{u}} \quad (5-26)$$

这个就是得到的伴随方程。这个方程与有限元的基本方程 $\mathbf{K}\mathbf{u} - \mathbf{f} = \mathbf{0}$ 格式上可以保持一致，在线性有限元问题中， $\mathbf{K}^T = \mathbf{K}$ ， \mathbf{K} 是对称阵；而在非线性有限元问题里， \mathbf{K} 不一定是对称阵。对于线性模型，由于 $\mathbf{K}^T = \mathbf{K}$ ，因此我们称系统是自伴随的。右手边的向量 $\frac{\partial I}{\partial \mathbf{u}}$ 我们称为伪载荷。由于在求解有限元模型的位移时已经对 \mathbf{K} 做了 LU 或 Cholesky 分解，因此在这一步可以大大简化运算量。

通常现有的高效的数值优化算法需要用到二阶导数（即黑塞矩阵）的信息。令

$$\mathbf{K} \frac{d\mathbf{u}}{d\mathbf{p}} = \frac{\partial \mathbf{F}}{\partial \mathbf{p}} - \frac{\partial \mathbf{K}}{\partial \mathbf{p}} \equiv \mathbf{R}_v \quad (5-27)$$

$$\frac{d^2 \mathbf{u}}{d\mathbf{p}^2} = \frac{\partial \mathbf{R}_v}{\partial \mathbf{p}} + \frac{\partial \mathbf{u}}{\partial \mathbf{p}} \quad (5-28)$$

$$\frac{d^2 I}{d\mathbf{p}^2} = \frac{\partial^2 I}{\partial \mathbf{p}^2} + 2\left(\frac{\partial^2 I}{\partial \mathbf{u} \partial \mathbf{p}}\right)^T \frac{d\mathbf{u}}{d\mathbf{p}} + \left(\frac{\partial I}{\partial \mathbf{u}}\right) \frac{d^2 \mathbf{u}}{d\mathbf{p}^2} \quad (5-29)$$

将式 (5-26) 带入到上式中，可以得到二阶导数的表达式为

$$\frac{d^2 I}{d\mathbf{p}^2} = \frac{\partial^2 I}{\partial \mathbf{p}^2} + 2\left(\frac{\partial^2 I}{\partial \mathbf{u} \partial \mathbf{p}}\right)^T \frac{d\mathbf{u}}{d\mathbf{p}} + \psi^T \left(\frac{\partial \mathbf{R}_v}{\partial \mathbf{p}} + \frac{\partial \mathbf{R}_v}{\partial \mathbf{p}} \frac{d\mathbf{u}}{d\mathbf{p}}\right) \quad (5-30)$$

5.5 伴随方法与气动力方程结合

根据式 (3-16)，残差方程为：

$$A = \begin{bmatrix} A_E - e \\ e^T & 0 \end{bmatrix} \begin{bmatrix} \Gamma \\ \alpha \end{bmatrix} - \begin{bmatrix} \gamma_{jig} + \gamma_\Delta \\ nW/qb^2 \end{bmatrix} \quad (5-31)$$

令 $w^T = [\Gamma \quad \alpha]$, 有

$$\frac{\partial A^T}{\partial w} \psi = -\frac{\partial I}{\partial w} \quad (5-32)$$

其中, $v = -(\gamma_{jig} + \gamma_\Delta + \alpha e)$ 则可以得到气动力方程的伴随向量为

$$\frac{\partial A}{\partial w} = \begin{bmatrix} \frac{\partial A}{\partial \Gamma} & \frac{\partial A}{\partial \alpha} \end{bmatrix} = \begin{bmatrix} A_E - e \\ e^T & 0 \end{bmatrix} \quad (5-33)$$

具体的, 可以写成

$$\frac{\partial A}{\partial p} = \begin{bmatrix} \frac{\partial A}{\partial \gamma_{jig}} & \frac{\partial A}{\partial \Lambda} \end{bmatrix} = \begin{bmatrix} -I & \frac{\partial A}{\partial \Lambda} \\ \mathbf{0} & 0 \end{bmatrix} \quad (5-34)$$

5.6 具体指标的梯度计算

5.6.1 应力的梯度

根据材料力学知识和有限元分析, 应力的计算表达式为

$$\sigma = Su \quad (5-35)$$

因此根据式 (5-25), 有应力的梯度表达式为

$$\frac{d\sigma}{dp} = \frac{\partial \sigma}{\partial p} - \psi^T \frac{\partial R}{\partial p} = \frac{\partial S}{\partial p} u - \frac{\partial K}{\partial p} u \quad (5-36)$$

而相应的伴随方程为

$$K^T \psi = \frac{\partial \sigma}{\partial u} = S \quad (5-37)$$

5.6.2 KS 函数的梯度

在设计过程中, 最为重要的约束条件是结构的强度极限约束。考虑对于每个结构有限元, 有

$$g_m = 1 - \frac{\sigma_m}{\sigma_y} \geq 0 \quad (5-38)$$

其中, σ_m 是第 m 个有限元元素的 von Mises 应力, σ_y 是材料的屈服极限。根据此, 我们采用 KS 函数 (Kresisselmeier-Steinhauser function)^[9]:

$$KS = -\frac{1}{\rho} \ln \left[\sum_m e^{-\rho g_m} \right] \quad (5-39)$$

这个函数表示了所有应力不超过屈服应力的不等约束的下界， ρ 为一个正的参数，代表这个下界距离有多靠近实际约束。而 KS 函数的梯度有两种求法。首先，

$$\frac{\partial KS}{\partial \sigma_i} = \frac{\partial KS}{\partial g_i} \frac{\partial g_i}{\partial \sigma_i} \quad (5-40)$$

$$= \left[\sum_i e^{-\rho g_i} \right]^{-1} e^{-\rho g_i} \cdot \left(-\frac{1}{\sigma_y} \right) \quad (5-41)$$

$$= -[\sigma_y \sum_i e^{-\rho g_i}]^{-1} e^{-\rho g_i} \quad (5-42)$$

而 KS 函数对设计变量的偏导数可以写成

$$\frac{dKS}{d\mathbf{p}} = \sum_i \frac{\partial KS}{\partial \sigma_i} \frac{d\sigma_i}{d\mathbf{p}} \quad (5-43)$$

这样就可以将式 (5-36) 得到的结果带入到式 (5-43) 中计算。也可以再次列出一个伴随方程

$$\mathbf{K}^T \boldsymbol{\psi} = \frac{\partial KS}{\partial \mathbf{u}} \quad (5-44)$$

式 (5-36) 的右手边的表达式为

$$\frac{\partial KS}{\partial \mathbf{p}} = \sum_i \frac{\partial KS}{\partial \sigma_i} \frac{\partial \sigma_i}{\partial \mathbf{p}} \quad (5-45)$$

$$= \sum_i \left\{ [\sigma_y \sum_i e^{-\rho g_i}]^{-1} e^{-\rho g_i} \frac{\partial S_i}{\partial \mathbf{p}} \mathbf{u} \right\} \quad (5-46)$$

得到 KS 函数梯度的表达式为

$$\frac{dKS}{d\mathbf{p}} = \frac{\partial KS}{\partial \mathbf{p}} - \boldsymbol{\psi}^T \frac{\partial \mathbf{R}}{\partial \mathbf{p}} = \frac{\partial KS}{\partial \mathbf{p}} \mathbf{u} - \frac{\partial \mathbf{K}}{\partial \mathbf{p}} \mathbf{u} \quad (5-47)$$

5.7 计算效率对比

设某段矩形截面悬臂梁，长度为 1 米，截面宽度为 1 米，厚度为 1 米，弹性模量为 6Pa，受到集度为 1N/m 的均布载荷。设材料的屈服应力为 1.6Pa，KS 函数的参数 $\rho = 1$ 。将梁等分为若干个单元，各个单元的截面厚度为设计变量。通过伴随方法和有限差分法分别计算梁的应力和 KS 函数关于截面厚度的梯度，有限差分法计算的相对步长为 %1。二种方法的计算效率对比如图所示。可以看出，无论是求解

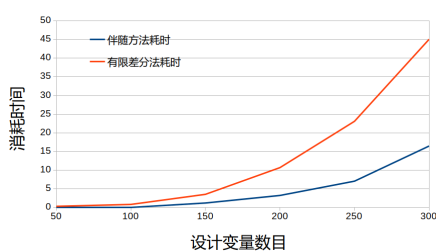


图 5-1 应力梯度计算耗时对比

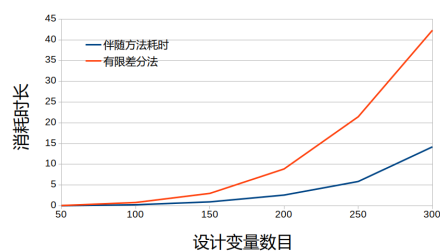


图 5-2 KS 函数梯度计算耗时对比

Fig. 5-1 Performance Comparison of Calculating Stress Gradients Fig. 5-2 Performance Comparison of Calculating KS Function Gradients

应力的梯度还是 KS 函数的梯度，有限差分法的计算耗时大约是伴随方法的 3 倍。这里计算 KS 函数的梯度时采用式 (5-43)，但实际上式 (5-43) 与式 (5-47) 两种计算方式并无明显计算效率差别。这个问题存在解析解，通过与解析解的对比，发现伴随方法计算梯度不仅速度更快，而且精度更高。

5.8 小结

本章介绍了几种常用的数值求解导数的方法，比较其各自的精度和效率，并重点介绍伴随方法，推导了一般的线性方程和非线性方程的伴随方法，并结合有限元和片条理论推导各自的伴随方程。根据梁的应力的实际问题，通过有限元的伴随方程和有限差分法分别求导数记录计算耗时，并与解析解对比，验证了伴随方法求导数在精度和速度两方面都具有优势。

第 6 章 机翼优化实例

6.1 引言

对于一个需要优化的机翼，需要按照前文研究的方法对其进行流固耦合分析，并采用适合的方法进行优化。本章将研究流固耦合方程的伴随方程，并推导梯度的表达式，对比对于流固耦合方程伴随方法与有限差分法求梯度的耗时，选取优化指标与约束条件，并研究合适的优化算法对性能指标进行优化，并分析优化得到的结果。

6.2 流固耦合方程中梯度的计算

设某段矩形机翼翼展为 5 米，展弦比为 9，机翼梁的为圆柱形实心梁，梁的直径为 0.06cm，梁的材料的弹性模量为 2GPa，泊松比为 0.2，材料的屈服应力为 $\sigma_m = 3 \times 10^6 \text{Pa}$ ，梁位于弦长的 30% 处。假设空气密度为 $\rho = 1.225 \text{kg/m}^3$ ，风速为 18m/s，机翼对气流的攻角为 5° ，机翼的升力线斜率为 5/rad. 设计变量为不同截面梁的直径。

那么式 (4-2) 和式 (4-1) 的伴随方程可以写成分块矩阵的形式为

$$\begin{bmatrix} \frac{\partial R}{\partial \mathbf{w}} & \frac{\partial R}{\partial \mathbf{u}} \\ \frac{\partial S}{\partial \mathbf{w}} & \frac{\partial S}{\partial \mathbf{u}} \end{bmatrix}^T \begin{bmatrix} \psi \\ \phi \end{bmatrix} = - \begin{bmatrix} \frac{\partial I}{\partial \mathbf{w}} \\ \frac{\partial I}{\partial \mathbf{u}} \end{bmatrix} \quad (6-1)$$

具体各个偏导数的表达式为：

$$\frac{\partial \mathbf{R}}{\partial \mathbf{w}} = \mathbf{A} \quad (6-2)$$

$$\frac{\partial \mathbf{R}}{\partial \mathbf{u}} = \begin{bmatrix} \mathbf{0}_{n \times 2n} & -\mathbf{I}_{n \times n} \end{bmatrix} \quad (6-3)$$

$$\frac{\partial S}{\partial \mathbf{w}} = -\frac{\partial f}{\partial \mathbf{w}} \quad (6-4)$$

$$\frac{\partial S}{\partial \mathbf{u}} = \mathbf{K} \quad (6-5)$$

由于性能指标的分子分母的表达式分别为

$$W = c(r_1^2 + r_2^2 + \dots) \quad (6-6)$$

$$L = qd(w_1 + w_2 + \dots) \quad (6-7)$$

因此式 (6-1) 的右边可以写为

$$\frac{\partial I}{\partial \mathbf{u}} = \mathbf{0}_{3n \times n} \quad (6-8)$$

$$\frac{\partial I}{\partial \mathbf{w}} = \begin{bmatrix} -\frac{qdW}{L^2} \\ -\frac{qdW}{L^2} \\ \vdots \\ -\frac{qdW}{L^2} \end{bmatrix} \quad (6-9)$$

von Mises 应力:

$$\sigma_m = \frac{1}{\sqrt{2}} \sqrt{(\sigma_x - \sigma_y)^2 + (\sigma_y - \sigma_z)^2 + (\sigma_z - \sigma_x)^2 + 6(\tau_{xy}^2 + \tau_{yz}^2 + \tau_{zx}^2)} \quad (6-10)$$

在这里可以简化为

$$I = \sigma_m = \frac{1}{\sqrt{2}} \sqrt{\sigma^2 + 3\tau^2} \quad (6-11)$$

$$\frac{\partial I}{\partial \mathbf{u}} = \frac{\partial \sigma_m}{\partial \sigma} \frac{\partial \sigma}{\partial \mathbf{u}} + \frac{\partial \sigma_m}{\partial \tau} \frac{\partial \tau}{\partial \mathbf{u}} \quad (6-12)$$

$$= \frac{\sigma}{\sqrt{\sigma^2 + 3\tau^2}} \mathbf{S}_1 + \frac{3\tau}{\sqrt{\sigma^2 + 3\tau^2}} \mathbf{S}_2 \quad (6-13)$$

设性能指标 I 为

$$I = \frac{W}{L} \quad (6-14)$$

性能指标对设计变量导数的表达式为

$$\frac{dI}{d\mathbf{p}} = \frac{\partial I}{\partial \mathbf{p}} + \boldsymbol{\psi} \frac{\partial \mathbf{R}}{\partial \mathbf{p}} + \boldsymbol{\phi} \frac{\partial \mathbf{S}}{\partial \mathbf{p}} \quad (6-15)$$

$$\frac{\partial I}{\partial \mathbf{p}} = \begin{bmatrix} \frac{2cr_1}{L} \\ \frac{2cr_2}{L} \\ \vdots \\ \frac{2cr_n}{L} \end{bmatrix} \quad (6-16)$$

$$\frac{\partial \mathbf{R}}{\partial \mathbf{p}} = \mathbf{0}_{n \times n} \quad (6-17)$$

$$\frac{\partial S}{\partial \mathbf{p}} = \frac{\partial \mathbf{K}}{\partial \mathbf{p}} \mathbf{u} \quad (6-18)$$

在这里如果推导 $\frac{\partial \mathbf{K}}{\partial \mathbf{p}}$ 的表达式，由于是一个三阶张量，极易出错且不易直接写成书面形式，不在这里写出。最终对比伴随方法和有限差分法计算梯度的效率，不同于纯有限元方程时有限差分法大概是伴随方法 3 倍的情况，在流固耦合的方程下，由于每一次求解流固耦合方程都需要耗费大量时间，因此运用伴随方法求解梯度带来了显著的效益，如图6-1所示，而且设计变量越大，效率差异越明显。事实上，当设计变量数目为 200 时，运行 10 次伴随方法求梯度的耗时为 15.8884 秒，而 10 次有限差分法耗时为 813.904 秒。如果求解气动力的方法改为更为复杂的涡格法甚至是 CFD 方法，那么伴随方法将会带来更大的收益^[9]。

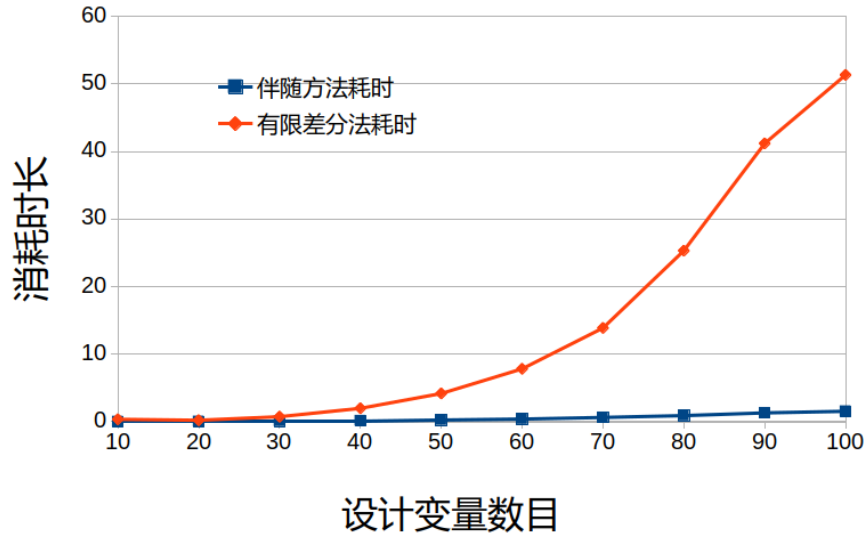


图 6-1 流固耦合方程的两种方法耗时对比

Fig.6-1 Cost Comparison in Aero-Structural Coupled Equations

6.3 机翼的优化结果

在这里给出如下一个带约束的优化问题：

$$\begin{aligned} &\text{minimize} && I = \frac{L}{W} \\ &\text{w.r.t.} && \mathbf{p} \\ &\text{s.t.} && KS > 0 \end{aligned} \quad (6-19)$$

和之前所给的条件一致, 设某段矩形机翼翼展为 5 米, 展弦比为 9, 机翼梁的为圆柱形实心梁, 梁的直径为 0.06cm, 梁的材料的弹性模量为 2GPa, 泊松比为 0.2, 材料的屈服应力为 $\sigma_m = 3 \times 10^6 \text{Pa}$, 梁位于弦长的 30% 处。假设空气密度为 $\rho = 1.225 \text{kg/m}^3$, 风速为 18m/s, 机翼对气流的攻角为 5° , 机翼的升力线斜率为 5/rad. 设计变量为不同截面梁的直径。约束条件为 KS 函数需要大于 0, 优化过程需要最小化重量与升力之比。

许多带约束的优化问题通常采用序列二次规划算法。但是由于序列二次规划算法有大量的矩阵运算, 这样导致了存储复杂度为 $O(n^2)$, 时间复杂度为 $O(n^3)$, 因此对于有大量设计变量的问题不适用。文献^[16]提出了一种基于梯度的局部优化方法: 移动渐近线方法, 利用一阶导数信息求解带约束的优化问题。文献^[16]给出的是保守凸优化分离估计的提升形式, 在拓扑优化问题中得到了广泛的应用。这种算法能够保证收敛于局部最小值, 但不一定收敛于全局最小值。在每一个点 x , 移动渐近线方法计算函数 f 和约束方程的梯度, 并加上一个平方“惩罚”项以使得这个导数的近似值为保守的, 即位于实际函数的上方。这样做的目的是让函数能够是凸的。

在 GitHub 上有开源的 C++ 非线性数值优化代码库 NLOPT, 将源码下载下来并自行编译即可在自己的程序中使用。NLOPT 提供了大量的高效数值优化算法, 包括局部优化算法, 全局优化算法, 无梯度优化算法, 基于梯度的优化算法, 可以处理各种不带约束或带约束优化问题。这里使用 NLOPT 的移动渐近线算法, 需要向程序库提供性能指标以及一阶导数, 约束条件以及一阶导数。通过伴随方法求出性能指标以及 KS 函数的一阶导数, 借助于 NLOPT, 选取初始条件为 $t = 0.06e$, KS 函数的参数 $\rho = 100$, 初始的机翼重量为 0.036, 而产生的升力为 128.306N, 优化后得到的结果为升力 $L = 131.472\text{N}$, 升力提升了 2.47%, 而重量为 0.02756456959, 重量减少了 23.43%, 优化后的机翼的厚度分布如图6-2所示, 可以看出翼根较厚而翼梢较薄。

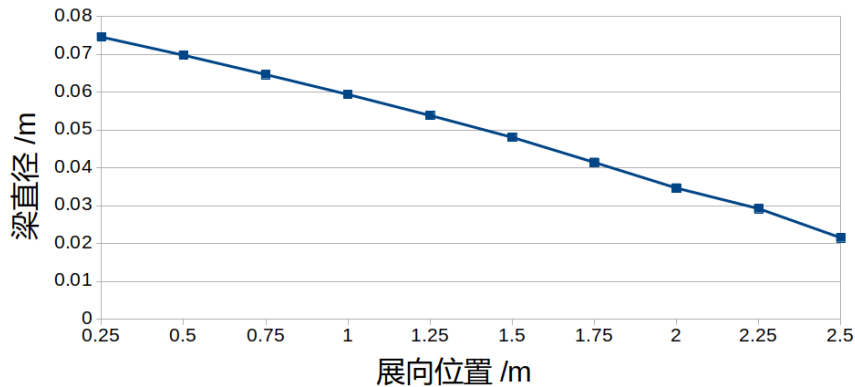


图 6-2 厚度沿展向的分布

Fig.6-2 Beam's Thickness Distribution along with the Wing Span

优化前性能指标 $I = 0.000280579$ ，优化后得到的性能指标 $I = 0.0002096604815$ ，优化后的性能指标下降了 25.28%。优化前后的应力沿展向的分布如图6-3所示，可以看出，在机翼的根部和中部的应力接近屈服应力，而翼梢处应力较小因为这里弯矩很小，应力分布没有超过强度极限，且大致满足满应力准则，并尽可能的减少了质量。

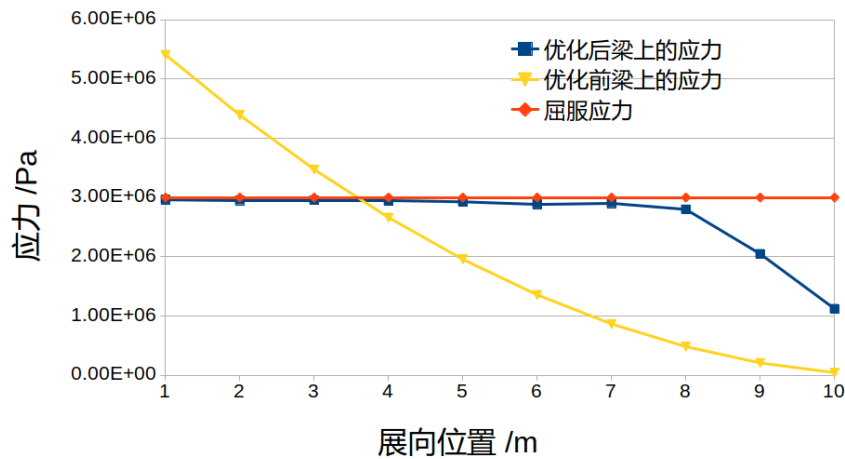


图 6-3 应力沿展向的分布

Fig.6-3 Beam's Stress Distribution along with the Wing Span

优化后前弹性机翼的扭转角如图6-4所示。可以看出，优化后的机翼在翼尖处有更大的弹性扭转角，因此优化后的机翼产生了更多的升力。

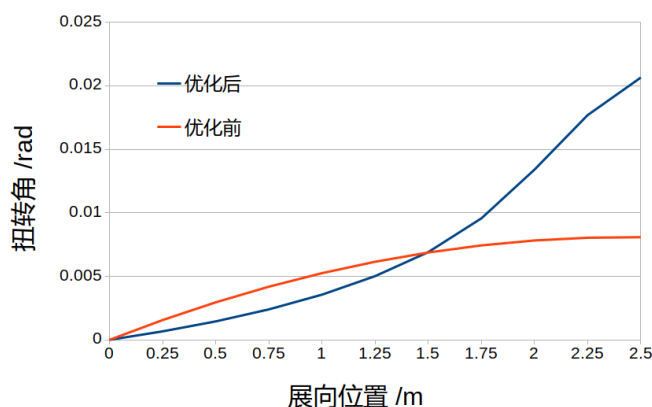


图 6-4 扭转角沿展向的分布

Fig.6-4 Wing's Deflection Angle along with the Wing Span

值得注意的是参数 ρ 的选取, 太大或者太小都不行, 太大会使最终的应力过于远离屈服应力, 太小则会使计算过程缓慢, 因此需要反复选取。在这里是通过试凑的方式使得优化的结果较为满意, 既能够最小化指标又能够尽可能接近屈服应力。多次试验表明, 在这个优化问题下 ρ 取到几百的数量级能够优化出较好的结果。

6.4 小结

本章推导了在流固耦合方程下的一些具体指标的伴随方程以及梯度关于伴随向量的表达式。通过流固耦合求解器, 使用 NLOPT 库中的移动渐进线算法, 基于伴随方法求性能指标关于设计变量的导数, 高效地优化得到了满足应力约束下的静气动弹性问题下的最优机翼。可以看出, KS 函数约束在 ρ 值选取合适时, 能够很好的表示所有单元处的应力约束, 使得优化的结果尽可能接近于满屈服应力状态。优化后的机翼主梁厚度呈现根部厚而梢部薄的特点, 使得重量尽可能小的情况下升力尽可能大。此外, 由于无论是序列二次规划算法还是移动渐进线算法优化得到的结果均是局部的, 因此后续工作中可以先使用一些全局搜索策略, 再在局部进行基于梯度的优化。

结 论

以对考虑气动弹性特性的机翼为目标，本文开展了以下研究工作：

（1）研究有限元的基本理论和工程梁的有限元模型，编写求解弯曲单元、扭转单元和平板单元的有限元求解器；

（2）研究片条理论、数值升力线方法和涡格法等工程常用的气动特性计算方法，并编写相应的求解器；

（3）研究针对气动弹性耦合方程的迭代算法，并编写基于非线性分块高斯赛德尔算法的求解器对实际的机翼在气流下的变形和升力进行计算；

（4）研究一般的伴随方程，并列出有限元方程的伴随方程，对一段梁的应力和KS函数分别用伴随方法和有限差分法求梯度，对二者的计算效率进行了对比；

（5）研究流固耦合方程的伴随方程，给出一个实际的机翼模型并提出性能指标，用伴随方法求解梯度并与有限差分法求梯度的耗时进行对比，并对机翼的性能指标通过移动渐进线方法进行优化，得到优化后的机翼。

在研究过程中，这项工作仍然有值得进一步可以改进的地方，列举如下：

（1）在这里仅仅考虑的是静力学的问题，而对于整个飞机而言，涉及结构动力学、飞行力学等方面的优化问题有很大的研究价值；

（2）对于大展弦比机翼，通常具有很强的非线性，如何针对非线性方程列出伴随方程并求解仍然是个问题；

（3）对于有限元模型，刚度矩阵实际为稀疏矩阵，而在自己编写的求解器中未将刚度矩阵处理为稀疏矩阵，对于稀疏矩阵，合适的存储方式、求解方式能够减小计算机的运行开销，从而实现求解更大自由度的问题；

（4）对于许多这里涉及的许多导数项的具体形式，涉及的雅阁比矩阵形式复杂，如果是三阶张量就更为复杂，人工推导表达式容易出错，应考虑采用自动微分方法，让计算机进行链导法则的运算，减小出错的可能性；

（5）事实上，无论是有限元还是气动计算都有已经成熟的开源代码库，例如有限元的 deal.ii，流体力学的 openfoam，流固耦合求解器 preCICE，以及自动微分运算规则库 adept，对于已有的开源程序库，应该多多利用而非自己闭门造车。

参考文献

- [1] 陈颂. 基于梯度的气动外形优化设计方法及应用[D]. [S.l.]: 西北工业大学, 2016.
- [2] 茅志颖. 结构动力学设计渐进优化方法研究[D]. [S.l.]: 南京航空航天大学, 2010.
- [3] Garg N, Kenway G K W, Martins J R R A, et al. High-fidelity multipoint hydrostructural optimization of a 3-D hydrofoil[J]. Journal of Fluids & Structures, 2017, 71 : 15 – 39.
- [4] Sorensen D C. Newton's Method with a Model Trust Region Modification[J]. Siam Journal on Numerical Analysis, 1982, 19(2) : 409 – 426.
- [5] Schittkowski K, Yuan Y X. Sequential Quadratic Programming Methods[M]. [S.l.]: Springer New York, 2012 : 147 – 224.
- [6] Terlaky T, Boggs P T. Interior-point methods[J]. Encyclopedia of Operations Research & Management Science, 2001, 6(4) : 402 – 411.
- [7] Martins J R R A, Sturdza P, Alonso J J. The complex-step derivative approximation[J]. Acm Transactions on Mathematical Software, 2003, 29(3) : 245 – 262.
- [8] Lyu Z, Kenway G K, Paige C, et al. Automatic Differentiation Adjoint of the Reynolds-Averaged Navier-Stokes Equations with a Turbulence Model[C] // AIAA Computational Fluid Dynamics Conference. 2013.
- [9] Martins J R R A, Alonso J J, Reuther J J. A Coupled-Adjoint Sensitivity Analysis Method for High-Fidelity Aero-Structural Design[J]. Optimization & Engineering, 2005, 6(1) : 33 – 62.
- [10] Errico R M. What Is an Adjoint Model?.[J]. Bulletin of the American Meteorological Society, 1997, 78(11) : 2577 – 2591.
- [11] 黄江涛. 飞行器气动/结构多学科延迟耦合伴随系统数值研究[J/OL]. 航空学报, 2018, 39(5) : 121731. http://hkxb.buaa.edu.cn/CN/abstract/article_17009.shtml.
- [12] Anderson J D. Fundamentals of Aerodynamics[M]. [S.l.]: McGraw-Hill,, 1984.
- [13] 陈桂彬. 气动弹性设计基础[M]. [S.l.]: 北京航空航天大学出版社, 2010.

- [14] Anderson J D, Corda S. Numerical lifting line theory applied to drooped leading-edge wings below and above stall[J]. Journal of Aircraft, 1980, 17(12): 898–904.
- [15] Kennedy G J, Martins J. Parallel solution methods for aerostructural analysis and design optimization[C] //Proceedings of the 13th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference, Fort Worth, TX. 2010.
- [16] Svanberg K. A Class of Globally Convergent Optimization Methods Based on Conservative Convex Separable Approximations[J]. Siam Journal on Optimization, 2002, 12(2): 555–573.

哈尔滨工业大学本科毕业设计（论文）原创性声明

本人郑重声明：在哈尔滨工业大学攻读学士学位期间，所提交的毕业设计（论文）《基于伴随方法的机翼气动弹性优化设计》，是本人在导师指导下独立进行研究工作所取得的成果。对本文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明，其它未注明部分不包含他人已发表或撰写过的研究成果，不存在购买、由他人代写、剽窃和伪造数据等作假行为。

本人愿为此声明承担法律责任。

作者签名：

日期： 年 月 日

致 谢

这次毕业设计基本在北航完成，我衷心感谢杨超老师和谢长川老师，在北航我获得了许多理论上的指导，多亏老师渊博的知识和细心的指导，言传身教将使我终生受益。以及在北航提供的良好的实验室的学习氛围。感谢实验室的安朝师兄，孟杨师兄，茂森师兄，杨澜师姐，吴越师兄，张桢锴师兄，高敏师兄，张志涛师兄，李应杰师兄，刘晨宇师兄，以及孙萌，胡天豪同学，为我在北航提供了各种帮助。也要衷心感谢衷心感谢哈工大的导师李顺利教授对本人的精心指导。还要感谢我的哈工大所有的同学，其中最重要的是我的室友张伟涛，李坤原，徐宇，张浩然和李溪远。我也要感谢我的父母亲人们，总是一直在关心我。最后还要特别感谢我的女朋友黄春蕾，在我最好的时候遇见了你。

附录 1 外文资料的调研阅读报告或书面翻译

基于耦合伴随方程的敏度的气动结构优化

摘要：一种新的子空间优化方法介绍了执行航空结构设计。该方法依赖于半解析伴随方法包括后最优性的敏感性分析灵敏度信息来自结构优化子问题。由此产生的耦合后最优性灵敏度方法用于指导基于梯度的方法优化算法。新方法简化了系统级的问题，从而减少了对潜在昂贵的空气动力学的调用数量求解。飞机的航空结构优化机翼采用线性气动和结构进行分析并进行性能比较新方法和传统的多学科之间可行的方法。新的不对称次优化方法被发现是更有效的当它充分减少数量时系统评估或足够大时纪律解决时间之间的差异。

1.1 引言

1.1.1 MDO 架构

在 multidisciplinary optimization (MDO) 领域人们已经开展了大量的研究，并将其运用在飞行器设计上。Sobieszcanski-Sobieski 和 Haftka 的调研论文提供了一个综合的关于这项工作的全貌，展示了在 MDO 问题所引发的一系列解决策略。一个最广泛的 MDO 技术就是气动和结构耦合的优化，因为这二者的交互是飞行器设计考虑的最主要因素。

MDO 的本质是要去解决在组织和计算复杂度上的挑战，有许多种不同的 MDO 架构来处理这种复杂度，可以在 Tedford 和 Martins(2006) 看见它们之间的比较。这些架构可以分为两大类：这些体系结构可以分为两大类：单层配方和多层配方。单级制定，如多学科可行性 (MDF) 和单个学科可行性体系结构，强加了一个单一的系统级优化器，它可以控制整个系统的状态 (Cramer et al. 1994; Tribes et al. 2005)。包括协同优化 (Kroo 1997; Braun 等人 1996) 和双层综合系统合成 (BLISS; Sobieszcanski-Sobieski et al. 1998; Kodiyalam 和 Sobieszcanski-Sobieski 2002) 的多级配方将原始问题划分为更小的子问题优化。

在这项研究中，我们开发了一个涉及子空间优化的新型混合架构。目标是降低解决 MDO 问题的成本，这些问题在惩罚性解决方案时间之间存在很大差异，高通量飞机结构优化通常就是这种情况 (Martins et al. 2004)。良好收敛性的关键考虑是多学科敏感性的有效计算。为了实现这一点，耦合伴随方法 (Martins et al.

2005) 被扩展到包括次空间优化的后最优信息。

1.1.2 动力

飞机机翼阻力和结构重量之间的权衡取决于空气动力学和结构之间的两种主要相互作用。首先，结构重量影响所需的升力，这又影响升力引起的阻力。其次，空气动力学载荷会影响结构变形，从而改变空气动力学形状。因此，为了获得收敛的航空结构状态，需要两个学科的几次迭代。不同的研究人员已经解决了执行高

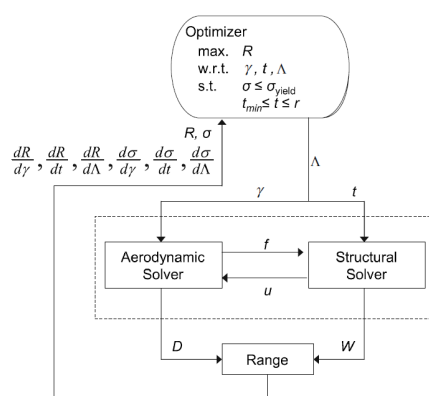


图 1-1 MDF 公式

Fig.1-1 MDF Formulation

保真航空结构分析和优化的挑战 (Maute et al.2001; Giunta 2000; Chattopadhyay and Pagaldipti 1995)。然而，由于涉及高计算成本，实际实施通常仅限于少数设计变量。解决如此大的优化问题的关键是耦合伴随方程的发展，这使得可以有效地计算随着设计变量的大量变化的梯度 (Martins et al.2005)。自那以后，有人提出了一些观察结果，从而产生了新的研究方向例如，为了利用伴随方法，Poon 和 Martins (2007) 开发了聚合压力约束的新方法。另一个问题是，涉及计算流体动力学求解器的空气动力学分析的计算成本比结构分析（线性有限元求解器）大一个数量级。

目前工作的目标是通过解决飞机结构分析模块中的结构子空间优化问题来利用这种计算失衡。图1-1显示了传统的 MDF 方法以及图1-2中提出的架构。在整篇文章中，汇集一组耦合学科的例程将被称为多学科分析 (MDA) 模块。因此，MDF 方法中的 MDA 涉及空气动力学和结构分析，而新方法中的 MDA 包括空气动力学分析和结构次优化。在所提出的架构中存在子空间优化意味着，对于每个空气动力学分析，必须执行结构优化，这将增加 MDA 的计算成本。然而，这种不对称的次优化方法的优点在于它通过重新定位所有结构设计变量和约束以及结构和规则来简化系统级问题。这减少了系统级优化器所需的梯度信息量，并且应该减少调

用次数到昂贵的空气动力学分析。这是在不影响空气动力学结构的影响的情况下完成的。这些学科之间的相互作用是精确建模的，并且分析的可靠性不受任何近似技术的限制，如响应表面，有时用于分层体系结构（Sobieski 和 Kroo 2000）。

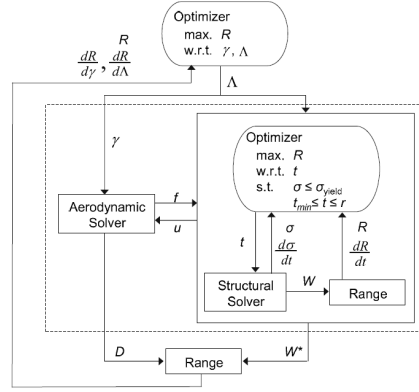


图 1-2 不对称的次优化方法

Fig. 1-2 Asymmetric Suboptimization Method

1.1.3 架构实现

对策优化方法的动机来源于学科解决方案时间差异导致的流体结构相互作用。然而，新制定的概念并不局限于航空结构优化，甚至不限于只有两种学科的系统。因此，新方法的实施细节将用一般符号来描述适用于更大和更复杂的设计应用。

以下讨论涉及具有 N 个学科的通用耦合系统。我们区分了只影响一个规则的局部设计变量 (x) 和影响规律的全局或共享设计变量 (z)。我们仅考虑在各学科之间交换的耦合变量 (y)。 x 和 y 变化规律具体以及耦合变量第 i 个纪律可以写成,

$$y_i = y_i(x_i, y_j, z) \quad (1-1)$$

其中 $j = 1, \dots, i-1, i+1, \dots, N$ 是对应于该学科所依赖的纪律的对应集合，因此 $j \neq i$ 。应该指出的是，就目前的讨论而言，本文通常用于表示向量的粗体符号已被索引符号暂时替代。当诸如 x 的变量被写入而没有下标时，它表示所有连接的 x_i 的向量。

不对称次优化架构背后的想法是，对于某些问题，将系统分为两类学科可能是有利的：仅分析学科 N_a 和执行优化的学科编号在指定学科时需要仔细考虑，因为只有分析计算上涉及较少，或者具有许多局部约束的那些适合作为次优化学科。在新的公式中，两组学科在 MDA 中耦合在一起，所有参与学科都包含在内，以便 $N_a + N_o = N$ 。

系统级优化问题可以表述为,

$$\begin{aligned}
 &\text{minimize} && F(z, x, y) \\
 &\text{w.r.t.} && z, x_{a_i} \\
 &\text{s.t.} && c_{a_i}(z, x_{a_i}, y_i) \geq 0 \quad i = 1, \dots, N_a
 \end{aligned} \tag{1-2}$$

x_{a_i} 和 c_{a_i} 是第 i 个仅分析专业的局部设计变量和约束条件。因此, 系统级问题可以控制仅分析学科的所有局部设计变量 x_a 和约束条件 c_a 。系统级优化器也会改变影响超过统计线 z 的设计变量。系统耦合性质的目标函数被指定为 F 。

MDA 模块包含所有的 N_a 和 N_o 规则。对于给定的一组设计变量和耦合变量输入, 每个只能分析的学科必须满足它们自己的控制方程。对于第 i 学科, 这可以写成,

$$\begin{aligned}
 &\text{given} && z, x_a, y_j \\
 &\text{solve} && R_i(z, x_{a_i}, y_j) \\
 &\text{to get} && y_i
 \end{aligned} \tag{1-3}$$

换句话说, 仅分析学科使用全局和局部设计变量以及任何所需的非局部耦合变量生成一组解决方案来满足 R_i 。

对子空间优化规则的每次调用不仅需要该学科的控制方程的解决方案, 还需要使用局部设计变量进行优化。第 i 个学科的子空间优化问题可以表述为,

$$\begin{aligned}
 &\text{given} && z, y_j \\
 &\text{minimize} && F(z, x, y) \\
 &\text{w.r.t.} && z, x_{o_i} \\
 &\text{s.t.} && c_{o_i}(z, x_{o_i}, y_i) \geq 0 \quad i = 1, \dots, N_a \\
 &\text{solve} && R_i(z, x_{o_i}, y_j) \\
 &\text{to get} && x_{o_i}, y_i
 \end{aligned} \tag{1-4}$$

其中 x_{o_i} 和 c_{o_i} 是第 i 个次优化规则的局部设计变量和约束条件。

所提议的 MDO 方法的收敛过程所涉及的步骤与 MDF 程序非常相似 (Cramer 等, 1994)。为所有全局和本地设计变量选择初始值。系统级优化器将 z 和 x_a 值传递给 MDA 模块, 其中包含软件分析的分析结果。使用块迭代过程来收敛贡献的学

科，牢记每个 N_o 规则的评估都需要单独的优化。MDA 考虑到每个学科分析所产生的耦合变量在相继的迭代中保持恒定在特定的容差范围内。这个收敛标准在数学上可以表述为，

$$y_i^{m+1} = y_i^m \pm \epsilon \quad i = 1, \dots, N \quad (1-5)$$

其中 y_i^m 表示 m 次迭代后第 i 个规则的耦合变量的值， ϵ 是期望的收敛容差。

一旦 MDA 收敛，就会计算系统级优化器所需的衍生产品。这些因素包括目标函数的总导数和系统级约束与仅分析设计变量 dF/dx_a 和 dc_a/dx_a 有关。这些导数可以使用耦合后优化灵敏度（CPOS）的方法进行计算，将在第 2 节介绍。一旦计算出来的敏感性，它们就会被传送到系统级优化器，以及 F 和 c_a 值，它们使用这些信息来确定下一个适当的设计步骤。随后，将更新后的 z 和 x_a 值返回给 MDA 模块，并重复该过程，直到系统级优化程序断定其已达到最佳设计。

1.1.4 气动结构模型

迄今为止我们所有的工作都集中在低成本的航空结构优化上，这在实施新的架构和灵敏度方法方面已被证明是有用的。空气动力学分析采用无粘面板代码模型，

$$A\Gamma - v = 0 \quad (1-6)$$

其中 A 是空气动力学影响系数矩阵， Γ 是面板循环的向量， v 是面板边界条件的向量，它仅仅是每个面板的局部迎角。如式 (1-6) 所示，空气动力学分析近似为线性系统。这与新架构背后的动机不一致，该架构旨在减少计算成本高昂的空气动力学求解器的评估次数。但是，在讨论第 3 节中的结果时要考虑到这一点。

空气动力学规则还强制机翼必须产生维持飞行所需的升力，其中 L 是机翼总升力， W 是飞机的总重量。

$$L - W = 0 \quad (1-7)$$

结构模型由一个单翼翼梁组成，该翼梁模型是用框架有限元素来表示管状翼梁。结构分析由以下等式决定，

$$Ku - f = 0 \quad (1-8)$$

其中 K 是结构的刚度矩阵， u 是位移矢量， f 是外力矢量。

选择 Breguet 范围方程来提供目标函数。这个表达式代表了飞机的阻力和结构重量之间的折衷，可以写成，

$$R = \frac{V}{c} \frac{L}{D} \ln \frac{W_i}{W_f} \quad (1-9)$$

其中 V 是巡航速度， c 是单位时间的燃料消耗， L/D 是升力与阻力的比率，而 W_i 和 W_f 是飞机的初始和最终重量。飞机的初始重量由结构重量和固定燃料重量组成，而最终重量仅仅是结构重量。这里的设计变量是机翼（ γ_{jig} ），管有限元（ t ）的壁厚和机翼扫掠（ Λ ）的双扭曲分布。

1.2 耦合后优化敏感度

灵敏度分析是执行基于梯度的优化时的一个重要考虑因素，因为衍生计算通常是优化周期内最昂贵的步骤。尽管存在结构优化程序，新结构看起来像是后优化灵敏度分析的逻辑应用。标准后优化分析允许改变最佳解决方案，而不必执行重新优化（Braun et al.1993），以改变先前固定的参数。不幸的是，这种方法没有考虑到手头的航空结构系统的耦合性质，并且很明显，这项工作需要扩展当前的理论。

耦合系统的敏感性可以用半分析方法进行计算，如 SobieszczanskiSobieski（1990）或耦合方法（Martins et al. 2005）引入的耦合直接灵敏度方程。这些方法允许计算系统级衍生物而无需解析 MDA，这大大降低了在整个系统分析中执行的有限差分的成本和不准确性。对于包含耦合气动残差（ A ）和结构残差（ S ），气压力学变量（ w ）和结构状态变量（ u ）的航空结构系统，耦合伴随方程可写为：

$$\begin{bmatrix} \frac{\partial A}{\partial w} & \frac{\partial A}{\partial u} \\ \frac{\partial S}{\partial w} & \frac{\partial S}{\partial u} \end{bmatrix} \begin{bmatrix} \psi \\ \phi \end{bmatrix} = - \begin{bmatrix} \frac{\partial F}{\partial w} \\ \frac{\partial F}{\partial u} \end{bmatrix} \quad (1-10)$$

其中 ψ 和 ϕ 分别是空气动力学和结构伴随向量。然后将这些伴随方程的解代入总灵敏度方程中，

$$\frac{dF}{dx} = \frac{\partial F}{\partial x} + \psi^T \frac{\partial A}{\partial x} + \phi^T \frac{\partial S}{\partial x} \quad (1-11)$$

找出系统级目标函数 F 关于系统级设计变量 x 而言的总导数。

在这种情况下，由于控制方程的解决方案，偏导数没有考虑到隐式依赖，而总导数确实包含这种隐式依赖。

1.2.1 气动残差

气动力学和结构残差很容易被识别。在动力学分析中，动力学方程解决了方程 (1-6) 和 (1-7) 中的问题，同时求解这两个方程得到的方程组的系统动力学方程可以写成：

$$\mathbf{A} = \begin{bmatrix} \mathbf{A} & -\mathbf{e} \\ \mathbf{e}^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\Gamma} \\ \alpha \end{bmatrix} - \begin{bmatrix} \boldsymbol{\gamma}_{jig} + \boldsymbol{\gamma}_{\Delta} \\ nW/qb^2 \end{bmatrix} = \mathbf{0} \quad (1-12)$$

每个面板的局部发生率已被扭曲，扭曲变形和攻角的个体贡献所取代，即 $\mathbf{v} = -(-\boldsymbol{\gamma}_{jig} + \boldsymbol{\gamma}_{\Delta} + \alpha\mathbf{e})$ 。第二行是一个表示升力约束的标量方程 (1-7)，其中 n 是面板的数量， q 是自由流动动压， b 是翼展。气动残差的状态变量是 $\mathbf{w}^T = [\boldsymbol{\Gamma}^T \quad \alpha]$ ，其中 α 是以弧度表示的飞机迎角。

1.2.2 结构残差

结构残差仅由式 (1-8) 给出，即，

$$\mathbf{S} = \mathbf{K}\mathbf{u} - \mathbf{f} = \mathbf{0} \quad (1-13)$$

结构残差的状态变量是位移 \mathbf{u} 。对于 MDF 体系结构，可以计算式 (1-12) 和 (1-13) 的耦合灵敏度，为单个优化问题提供梯度。然而，作为对称优化架构，MDA 模块中的结构规范不仅涉及分析，而且还包括在制定残差方程时必须考虑的全面优化。

1.2.3 结构优化残差

结构优化是通过改变内部机翼厚度，有效载荷分布来最大化飞机航程。优化限制在于防止结构应力 (σ) 超过材料屈服应力并保持厚度在最小标准值和半径之间，可以表示为，

$$\begin{aligned} &\text{maximize} && R \\ &\text{w.r.t.} && t \\ &\text{w.r.t.} && z, x_{oi} \\ &\text{s.t.} && \sigma_{\text{yield}} - \sigma \geq 0 \\ &&& t - t_{\min} \geq 0 \\ &&& r - t \geq 0 \end{aligned} \quad (1-14)$$

由于结构优化是一个受限制的，基于梯度的问题，优化器正在努力满足 Karush-

Kuhn-Tucker (KKT) 条件。KKT 条件必须满足结构最优，但它们并不完全描述次优化，因为最优也必须是结构解决方案。因此，将结构残差 (1-13) 与 KKT 条件相结合就完成了图像，并使我们能够充分表示结构优化残差：

$$\mathbf{O} = \begin{cases} \mathbf{K}\mathbf{u} - \mathbf{f} = \mathbf{0} \equiv \mathbf{O}_S \\ \frac{dR}{dt} + \lambda_{\sigma}^T \frac{d\sigma}{dt} + \lambda_t^T \mathbf{I} = \mathbf{0} \equiv \mathbf{O}_L \\ \sigma_{\text{yield}} - \sigma - S_{\sigma}^2 = \mathbf{0} \equiv \mathbf{O}_{\sigma} \\ t - t_{\min} - s_t^2 = \mathbf{0} \equiv \mathbf{O}_t \\ S_{\sigma} \lambda_{\sigma} = \mathbf{0}, s_t \lambda_t = \mathbf{0} \equiv \mathbf{O}_{s\lambda} \end{cases} \quad (1-15)$$

其中 $\lambda^T = [\lambda_{\sigma}^T \quad \lambda_t^T]$ 分别是应力和厚度约束的拉格朗日乘子，和 $s^T = [s_{\sigma}^T \quad s_t^T]$ 是与这些约束相关的松弛变量。KKT 方程忽略了半径约束，因为虽然它们被用来引导优化器远离非物理解，但它们对最优的解决方案是非活性的。总的衍生物在第一个 KK 条件下被提出，在优化残差 (1-15) 中表示为 \mathbf{O}_L ，因为找到了关于厚度设计变量的期望灵敏度需要结构方程的解决方案。这将在稍后更详细地讨论。整套优化状态变量由四个向量组成，因此 $\mathbf{y}^T = [\mathbf{u}^T \quad \mathbf{t}^T \quad \mathbf{s}^T \lambda^T]$ 。

对称优化方法的多学科分析可视为包含两个独立的学科：空气动力学和结构优化。在确定了结构优化残差 \mathbf{O} 之后，我们现在能够提出相应的耦合伴随方程，

$$\begin{bmatrix} \frac{\partial A}{\partial \mathbf{w}} & \frac{\partial A}{\partial \mathbf{y}} \\ \frac{\partial \mathbf{O}}{\partial \mathbf{w}} & \frac{\partial \mathbf{O}}{\partial \mathbf{y}} \end{bmatrix} \begin{bmatrix} \boldsymbol{\psi} \\ \boldsymbol{\zeta} \end{bmatrix} = - \begin{bmatrix} \frac{\partial F}{\partial \mathbf{w}} \\ \frac{\partial F}{\partial \mathbf{y}} \end{bmatrix} \quad (1-16)$$

其中 $\boldsymbol{\zeta}$ 是与结构子空间优化相关的伴随向量。相应的总灵敏度是，

$$\frac{dF}{d\mathbf{x}} = \frac{\partial F}{\partial \mathbf{x}} + \boldsymbol{\psi}^T \frac{\partial A}{\partial \mathbf{x}} + \boldsymbol{\zeta}^T \frac{\partial \mathbf{O}}{\partial \mathbf{x}} \quad (1-17)$$

回想一下，所提出的体系结构的系统级设计变量 (\mathbf{x}) 是每个面板 (γ_{jig}) 和整体的后掠 (Λ)，而在 MDF 方法中，它们也包括结构厚度 (\mathbf{t})。

计算新架构系统级灵敏度的另一种有效方法是相应的耦合直接灵敏度方程。直接和联合方法同时表示了部分微分项，但操作顺序是不同的。因此，使用直接方法计算敏感度的成本实际上与感兴趣函数的数量无关，而与伴随方法相关的成本本质上与设计变量的数量无关。由于新架构可以从系统级优化器中去除结构约束，因此唯一需要总导数的函数是距离方程。因此，使用伴随 CPOS 方法在不对称子优化方法中是特别有利的。

现在我们将更详细地描述 MDF 方法和新架构的耦合灵敏度方程的偏导数项。

1.2.4 气动灵敏度

气动残差相对于流量变量的偏导数可以分解为,

$$\frac{\partial A}{\partial \mathbf{w}} = \begin{bmatrix} \frac{\partial b_{m\Lambda}}{\partial \Gamma} \\ \frac{\partial A}{\partial \alpha} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & -\mathbf{e} \\ \mathbf{e}^T & 0 \end{bmatrix} \quad (1-18)$$

这是式 (1-12) 中所示的相同矩阵。

空气动力学残差相对于次优化状态变量的偏导数可以如下分解,

$$\frac{\partial A}{\partial \mathbf{y}} = \begin{bmatrix} \frac{\partial A}{\partial \mathbf{u}} & \frac{\partial A}{\partial t} & \frac{\partial A}{\partial s} & \frac{\partial A}{\partial \lambda} \end{bmatrix} = \begin{bmatrix} -\frac{\partial \gamma_{\Delta}}{\partial \mathbf{u}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 0 & -\frac{n}{qb^2} \frac{\partial W}{\partial t} & 0 & 0 \end{bmatrix} \quad (1-19)$$

KKT 状态的唯一直接影响是重量与结构厚度之间的关系。同时, 关于结构状态变量的偏导数只涉及每个面板的局部攻角, 这代表有限元位移如何转化为扭转变形。

我们写出关于系统级设计变量的敏感性,

$$\frac{\partial A}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial A}{\partial \gamma_{\text{jig}}} & \frac{\partial A}{\partial \Lambda} \end{bmatrix} = \begin{bmatrix} -\mathbf{I} & \frac{\partial A}{\partial \Lambda} \\ \mathbf{0} & 0 \end{bmatrix} \quad (1-20)$$

式 (1-19) 和 (1-20) 的偏导数是专门针对新架构而编写的。然而, MDF 灵敏度涉及相同的术语, 但重新排列的方式与其他方式不同。除了与 KKT 变量分组之外, $\frac{\partial A}{\partial \mathbf{u}}$ 单独编写, 并且 $\frac{\partial A}{\partial t}$ 被包括作为设计变量灵敏度。 $\frac{\partial A}{\partial s}$ 和 $\frac{\partial A}{\partial \lambda}$ 都是非双曲型, 两种结构所要求的空气动力学灵敏度之间基本没有差异。

1.2.5 结构分析灵敏度

结构方程式被包括作为结构优化残差的一部分。因此, 下面的敏感性术语被明确地写入 MDF 方法中, 并且将在后面的新结构的次优化敏感性中重复。

结构残差相对于流变量的敏感矩阵只涉及外力矢量

$$\frac{\partial S}{\partial \mathbf{w}} = \begin{bmatrix} \frac{\partial S}{\partial \Gamma} & \frac{\partial S}{\partial \alpha} \end{bmatrix} = \begin{bmatrix} -\frac{\partial f}{\partial \Gamma} & \mathbf{0} \end{bmatrix} \quad (1-21)$$

考虑设计和位移的灵敏度就是刚度矩阵

$$\frac{\partial S}{\partial \mathbf{u}} = \mathbf{K} \quad (1-22)$$

最后, 结构方程相对于系统级设计变量的偏导数是

$$\frac{\partial S}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial S}{\partial t} & \frac{\partial S}{\partial \gamma_{\text{jig}}} & \frac{\partial S}{\partial \Lambda} \end{bmatrix} = \begin{bmatrix} \frac{\partial \mathbf{K}}{\partial t} & \mathbf{0} & \frac{\partial \mathbf{K}}{\partial \Lambda} \mathbf{u} - \frac{\partial f}{\partial \Lambda} \end{bmatrix} \quad (1-23)$$

我们注意到在这种情况下跳汰机曲折不会直接影响结构。

1.2.6 结构优化灵敏度

如前所述，式 (1-15) 中的第一个 KKT 条件是用总的导数写出的，表明它可能需要求解一组残余方程。需要澄清的是，我们只是指结构残差，因为次优化是作为一个独立的学科运作的。由于航空器范围对厚度设计变量的依赖性通过结构权重项进入，并且因为重量仅仅是厚度的函数，明确地或以其他方式，因此，范围相对于厚度的总导数减小到一个偏导数。使用链式规则，这可以写成

$$\frac{dR}{dt} = \frac{\partial R}{\partial t} + \frac{\partial R}{\partial u} \frac{du}{dt} = \frac{\partial R}{\partial u} \frac{du}{dt} \quad (1-24)$$

总衍生物表示需要解析结构残差。因此，就状态变量而言， $\frac{du}{dt}$ 项取决于 $\mathbf{\Gamma}$ 和 t ，而应力约束本身只是位移的直接函数。

1.3 结果和讨论

在开展任何 MDO 试验之前，我们针对新的不对称次优化方法开发的 CPOS 方程式已根据有限差分方法进行了验证。这两种方法在表 1 中分别用五个曲面和五个设计变量进行比较，对应于五个面板和五个单元。第一块面板的夹具扭转固定。CPOS 方程中的偏导数使用复数阶导数近似计算 (Squire and Trapp 1998; Martin et al. 2003)，它提供了数值上的精确值。

如表 1 所示，CPOS 伴随结果与 5 位数的有限差异一致。在有限差分估计中，CPOS 的结果更加准确，可以作为减法消除误差。CPOS 方法的主要优点很明显：计算时间仅为有限差异所需时间的 3%。因此，CPOS 方法准确地模拟了我们的紧耦合系统的响应，而无需重新进行分析，甚至不会对子空间问题进行优化，从而比有限差分法提供显著的节省。

为了更好地理解 CPOS 伴随所涉及的计算成本，计算总衍生物的成本明细如表 2 所示。在本研究中，面板数量和有限元素数量均增加到 50。显然，计算偏导数是迄今为止的费用最低的方法，其中几乎所有的偏差都归因于 KKT 系统的偏导数。这并不奇怪，因为子空间优化导数包含二阶项。目前，这些高阶导数的计算是通过使用复合步法来确定一阶导数，然后在例程上执行有限差分以获得二阶导数。

偏导数项的计算超过伴随解的原因是因为当前的航空结构模型是用 Python 脚本语言编写的，而伴随向量是在预编译的数字包中使用线性求解运算来计算的。大多数大规模的计算要求使用 Fortran 等语言编译的语言。当对这些较高级别的系统

进行优化时，解决直接和伴随方法所需的方程组通常成为主要支出，并且使用伴随方法更加有利。如表 2 所示，将设计变量的数量从 5 增加到 50 并不会影响所需时间计算伴随矢量。唯一的成本增加是由于空气动力学残差对弹性扭转的依赖。

1.4 结论

我们开发了一种用于执行航空结构设计的小空间优化方法，该设计使用耦合的相邻感应方法来确定所需的系统级导数。我们发现，与传统的 MDF 方法相比，这种非对称的次优化方法提供了计算优势：它简化了系统级问题，从而减少了对 MDA 模块的调用次数，减少了总迭代次数，减少了对空气动力学分析的评估。

简单的外插值说明，新结构变得越来越有吸引力，因为空气动力学设计变量的数量或航空结构分析的可靠性都会增加。这可能是由于所开发的灵敏度方程式，其涉及包括耦合伴随灵敏度方程中的结构优化残差。

更一般地说，这里开发的伴随式 CPOS 方法可以应用于任何 MDO 问题，特别适用于在学科解算器的计算成本之间存在较大差异的问题。

附录 2 外文资料原文

Aero-structural optimization using adjoint coupled post-optimality sensitivities

Abstract: A new subspace optimization method for performing aero-structural design is introduced. The method relies on a semi-analytic adjoint approach to the sensitivity analysis that includes post-optimality sensitivity information from the structural optimization subproblem. The resulting coupled post-optimality sensitivity approach is used to guide a gradient-based optimization algorithm. The new approach simplifies the system-level problem, thereby reducing the number of calls to a potentially costly aerodynamics solver. The aero-structural optimization of an aircraft wing is performed using linear aerodynamic and structural analyses, and a performance comparison is made between the new approach and the conventional multidisciplinary feasible method. The new asymmetric suboptimization method is found to be the more efficient approach when it adequately reduces the number of system evaluations or when there is a large enough discrepancy between disciplinary solution times.

2.1 Introduction

2.1.1 MDO architecture

Extensive research has been conducted in the field of multidisciplinary design optimization (MDO) and its application to aircraft design. The survey paper by Sobieszczanski-Sobieski and Haftka (1997) provides a comprehensive overview of the work accomplished in this area, highlighting the different strategies that have been inspired by the inherent challenges of MDO. One of the most common applications of MDO techniques is coupled aerodynamic and structural (aero-structural) optimization, because the interaction between these two disciplines is a primary consideration in the design of aircraft.

The interdisciplinary coupling intrinsic to MDO tends to pose significant organizational and computational challenges, and there exist several different MDO architectures for dealing with this complexity, as compared by Tedford and Martins (2006). These architectures can be divided into two main classes: single-level formulations and multilevel

formulations. Single-level formulations, such as the multidisciplinary feasible (MDF) and the individual discipline feasible architectures, impose a single, system-level optimizer that is given control over the entire state of the system (Cramer et al. 1994; Tribes et al. 2005). Multilevel formulations, which include collaborative optimization (Kroo 1997; Braun et al. 1996) and bi-level integrated system synthesis (BLISS; Sobieszczanski-Sobieski et al. 1998; Kodiyalam and Sobieszczanski-Sobieski 2002), divide the original problem into smaller subproblem optimizations.

In this research, we develop a new hybrid architecture that involves a subspace optimization. The objective is to decrease the cost of solving MDO problems that exhibit a large discrepancy between disciplinary solution times, as is often the case in high-fidelity aerostructural optimization (Martins et al. 2004). The crucial consideration for good convergence properties is the efficient computation of multidisciplinary sensitivities. To accomplish this, the coupled adjoint method (Martins et al. 2005) is expanded to include postoptimality information from the subspace optimization.

2.1.2 Motivation

The trade-off between drag and structural weight for aircraft wings is governed by two main interactions between the aerodynamics and structures. First, the structural weight affects the required lift, which in turn affects the lift-induced drag. Second, the aerodynamic loads affect the structural deformations, which in turn change the aerodynamic shape. Therefore, to obtain a converged aero-structural state, several iterations of the two disciplines are needed.

Different researchers have addressed the challenge of performing high-fidelity aero-structural analysis and optimization (Maute et al. 2001; Giunta 2000; Chattopadhyay and Pagaldipti 1995). However, due to the high computational costs involved, practical implementations have generally been limited to only a few design variables. The key to solving such large optimization problems was the development of the coupled adjoint equations, which made it possible to efficiently compute gradients with respect to large numbers of design variables (Martins et al. 2005). Since then, several observations were made that resulted in new research directions. For instance, to take advantage of the adjoint approach, new ways to aggregate stress constraints were developed by Poon and Martins (2007). Another issue was the fact that the computational cost of the aerodynamic

analysis, which involves a computational fluid dynamics solver, was an order of magnitude greater than the structural analysis (a linear finite-element solver).

The goal of the present work is to take advantage of this computational imbalance by solving a structural subspace optimization problem within the aerostructural analysis module. The conventional MDF approach is shown in Fig. 1 along with the proposed architecture in Fig. 2. Throughout this paper, the routine that converges a coupled set of disciplines will be referred to as the multidisciplinary analysis (MDA) module. Therefore, the MDA in the MDF method involves the aerodynamic and structural analyses, whereas the MDA in the new method includes the aerodynamic analysis and the structural suboptimization. The presence of the subspace optimization in the proposed architecture means that, for each aerodynamic analysis, one has to perform a structural optimization, which will increase the computational cost of the MDA. The advantage of this asymmetric suboptimization method, however, is that it simplifies the system-level problem by relocating all of the structural design variables and constraints within the structural discipline. This reduces the amount of gradient information required by the system-level optimizer and should decrease the number of calls to the costly aerodynamic analysis. This is accomplished without having to compromise the influence of the structures on the aerodynamics. The interactions between the disciplines are modeled exactly, and the fidelity of the analyses is not limited by any approximation technique, such as a response surface, which are sometimes used in hierarchical architectures (Sobieski and Kroo 2000).

2.1.3 Architecture implementation

Motivation for the asymmetric suboptimization method originated from the discrepancy in disciplinary solution times apparent in fluid-structure interactions. However, the concept behind the new formulation is not limited to aero-structural optimization or even to systems with only two participating disciplines. Therefore, the implementation details of the new method will be described using general notation to account for larger and more complex design applications. The following discussion pertains to a general coupled system with N disciplines. We make a distinction between the local design variables (x) that only affect one discipline and the global or shared design variables (z) that affect more than one discipline. We also need to consider the coupling variables (y) that are exchanged between disciplines. Both x and y are discipline specific, and the coupling

variables for the i th discipline can be written as,

$$y_i = y_i(x_i, y_j, z) \quad (2-1)$$

where x_{a_i} and c_{a_i} are the local design variables and constraints of the i th analysis-only discipline. Therefore, the system-level problem is given control over all of the local design variables x_a and constraints c_a of the analysis-only disciplines. The system-level optimizer also varies the design variables that affect more than one discipline, z . The objective function that models the coupled nature of the system is designated as F .

The MDA module contains all of the N_a and N_o disciplines. For a given set of design variables and coupling variable inputs, each analysis-only discipline must satisfy their own governing equations. For the i th discipline, this can be written as,

$$\begin{array}{ll} \text{given} & z, x_a, y_j \\ \text{solve} & R_i(z, x_{a_i}, y_j) \\ \text{to get} & y_i \end{array} \quad (2-2)$$

In other words, the analysis-only discipline generates a set of solutions to satisfy R_i using the global and local design variables and any required non-local coupling variables.

Each call to a subspace optimization discipline requires not only a solution of the discipline's governing equations, but also an optimization using the local design variables. The subspace optimization problem for the i th discipline can be stated as,

$$\begin{array}{ll} \text{given} & z, y_j \\ \text{minimize} & F(z, x, y) \\ \text{w.r.t.} & z, x_{o_i} \\ \text{s.t.} & c_{o_i}(z, x_{o_i}, y_i) \geq 0 \quad i = 1, \dots, N_a \\ \text{solve} & R_i(z, x_{o_i}, y_j) \\ \text{to get} & x_{o_i}, y_i \end{array} \quad (2-3)$$

where x_{o_i} and c_{o_i} are the local design variables and constraints of the i th suboptimization discipline.

The steps involved in the convergence process of the proposed MDO approach closely resemble an MDF procedure (Cramer et al. 1994). Initial values are chosen for all of the

global and local design variables. The system-level optimizer passes the z and x_a values into the MDA module, containing all of the discipline analyses. A block-iterative procedure is used to converge the contributing disciplines, keeping in mind that each evaluation of an MD module requires a separate optimization. The MDA is considered to be converged once the coupling variables generated by each discipline analysis have remained constant to within a specified tolerance, over successive iterations. This convergence criteria can be stated mathematically as,

$$y_i^{m+1} = y_i^m \pm \epsilon \quad i = 1, \dots, N \quad (2-4)$$

where y_i^m represents the value of the i th discipline's coupling variables after m iterations, and ϵ is the desired convergence tolerance.

Once the MDA has converged, the derivatives required by the system-level optimizer are calculated. These include the total derivatives of the objective function and system-level constraints with respect to the analysis-only design variables, dF/dx_a and dc_a/dx_a . These derivatives can be efficiently calculated using the method of coupled post-optimality sensitivities (CPOS), which is introduced in Section 2. Once the sensitivities have been computed, they are passed to the system-level optimizer, along with the F and c_a values, which uses the information to determine the next appropriate design step. Subsequently, the updated z and x_a values are returned to the MDA module, and the process is repeated until the system-level optimizer concludes that it has reached the best possible design.

2.1.4 Aero-structural model

All of our work to date has focused on lower-fidelity aero-structural optimization, which has proven useful in implementing the new architecture and sensitivity method. The aerodynamic analysis employs an inviscid panel code to model the wing, which solves the system,

$$A\Gamma - \nu = 0 \quad (2-5)$$

where A is the aerodynamic influence coefficients matrix, Γ is the vector of panel circulations, and ν is the vector of panel boundary conditions, which is simply the local angle of attack of each panel. As indicated by (refesix), the aerodynamic analysis is being approximated as a linear system. This is not consistent with the motivation behind the new

architecture, which is to reduce the number of evaluations of a computationally expensive aerodynamic solver. However, this is taken into account when discussing the results in Section 3.

The aerodynamic discipline also enforces that the wing must produce the lift needed to maintain level flight, i.e.,

$$L - W = 0 \quad (2-6)$$

where L is the total wing lift and W is the total weight of the aircraft.

The structural model consists of a single wing spar, which is modeled using frame finite elements to represent a tube-shaped spar. The structural analysis is governed by the following equation,

$$\mathbf{K}\mathbf{u} - \mathbf{f} = \mathbf{0} \quad (2-7)$$

where \mathbf{K} is the stiffness matrix of the structure, \mathbf{u} is the displacement vector, and \mathbf{f} is the vector of external forces.

The Breguet range equation was selected to provide the objective function. This expression represents the trade-off between the drag and the structural weight of the aircraft, and can be written as,

$$R = \frac{V}{c} \frac{L}{D} \ln \frac{W_i}{W_f} \quad (2-8)$$

where V is the cruise velocity, c is the specific fuel consumption, L/D is the ratio of lift to drag, and W_i and W_f are the initial and final weights of the aircraft. The initial weight of the aircraft consists of the structural weight and a fixed fuel weight, whereas the final weight is simply the structural weight. The design variables that we use herein are the jig twist distribution of the wing (γ_{jig}), the wall thicknesses of the tube finite elements (\mathbf{t}), and the wing sweep (Λ)

2.2 Coupled post-optimality sensitivities (CPOS)

Sensitivity analysis is an important consideration when performing gradient-based optimization, as the derivative calculations are often the most costly step within the optimization cycle. Due to the presence of the structural optimization routine, the new architecture seemed like a logical application of post-optimality sensitivity analysis. Standard

post-optimality analysis allows for the change in the optimum solution with respect to a change in a previously fixed parameter to be attained, without having to perform a re-optimization (Braun et al. 1993). Unfortunately, this method does not take into account the coupled nature of the aero-structural system at hand, and it became clear that an extension of the current theory was needed for this work.

The sensitivities of coupled systems can be computed using semi-analytical methods, such as the coupled direct sensitivity equations introduced by Sobieszczanski- Sobieski (1990) or the coupled adjoint method (Martins et al. 2005). These methods allow for the system-level derivatives to be computed without having to resolve the MDA, which greatly reduces the cost and inaccuracy of finite-differencing performed on the entire system analysis. For the aero-structural system involving the coupled aerodynamic residuals (\mathbf{A}) and structural residuals (\mathbf{S}), as well as the aerodynamic state variables (\mathbf{w}) and the structural state variables (\mathbf{u}), the coupled adjoint equations can be written as,

$$\begin{bmatrix} \frac{\partial \mathbf{A}}{\partial \mathbf{w}} & \frac{\partial \mathbf{A}}{\partial \mathbf{u}} \\ \frac{\partial \mathbf{S}}{\partial \mathbf{w}} & \frac{\partial \mathbf{S}}{\partial \mathbf{u}} \end{bmatrix} \begin{bmatrix} \boldsymbol{\psi} \\ \boldsymbol{\phi} \end{bmatrix} = - \begin{bmatrix} \frac{\partial F}{\partial \mathbf{w}} \\ \frac{\partial F}{\partial \mathbf{u}} \end{bmatrix} \quad (2-9)$$

where $\boldsymbol{\psi}$ and $\boldsymbol{\phi}$ are the aerodynamic and structural adjoint vectors, respectively. The solution of these adjoint equations can then be substituted into the total sensitivity equation,

$$\frac{dF}{d\mathbf{x}} = \frac{\partial F}{\partial \mathbf{x}} + \boldsymbol{\psi}^T \frac{\partial \mathbf{A}}{\partial \mathbf{x}} + \boldsymbol{\phi}^T \frac{\partial \mathbf{S}}{\partial \mathbf{x}} \quad (2-10)$$

to find the total derivative of the system-level objective function, F , with respect to the system-level design variables, \mathbf{x} .

In this context, partial derivatives do not take into account the implicit dependence due to the solution of governing equations, whereas total derivatives do include this implicit dependence.

2.2.1 Aerodynamic residuals

The aerodynamic and structural residuals are easily identified. As the aerodynamic analysis involves solving (2-5) and (2-6), the aerodynamic residuals are the system of equations that result from solving those two equations simultaneously, and can be written as,

$$\mathbf{A} = \begin{bmatrix} \mathbf{A} & -\mathbf{e} \\ \mathbf{e}^T & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\Gamma} \\ \alpha \end{bmatrix} - \begin{bmatrix} \boldsymbol{\gamma}_{jig} + \boldsymbol{\gamma}_{\Delta} \\ nW/qb^2 \end{bmatrix} = \mathbf{0} \quad (2-11)$$

The local incidence for each panel has been replaced by the individual contributions of jig twist, twist deflection and angle of attack, i.e., $\mathbf{v} = -(-\boldsymbol{\gamma}_{jig} + \boldsymbol{\gamma}_{\Delta} + \alpha\mathbf{e})$, where \mathbf{e} is a vector of ones. The second row is a scalar equation that represents the lift constraint (2-6), where n is the number of panels, q is the free stream dynamic pressure, and b is the wing span. The state variables for the aerodynamic residuals are $\mathbf{w}^T = [\boldsymbol{\Gamma}^T \quad \alpha]$, where α is the angle of attack of the aircraft in radians.

2.2.2 Structural residuals

The structural residuals are given simply by (2-7), i.e.,

$$\mathbf{S} = \mathbf{K}\mathbf{u} - \mathbf{f} = \mathbf{0} \quad (2-12)$$

The state variables for the structural residuals are the displacements, \mathbf{u} . For the MDF architecture, the coupled sensitivities from (2-11) and (2-12) can be computed to provide gradients to a single optimization problem. However, for the asymmetric suboptimization architecture, the structural discipline within the MDA module not only involves an analysis, but also a full optimization that must be taken into account when formulating the residual equations.

2.2.3 Structural optimization residuals

The structural optimization is performed to maximize the aircraft range by varying the internal wing thicknesses, for a given load distribution. The optimization is constrained to prevent the structural stresses ($\boldsymbol{\sigma}$) from exceeding the yield stress of the material and to keep the thicknesses between a minimum gauge value and the radius of the spar and can be stated as,

$$\begin{aligned} &\text{maximize} && R \\ &\text{w.r.t.} && t \\ &\text{w.r.t.} && z, x_{o_i} \\ &\text{s.t.} && \sigma_{\text{yield}} - \sigma \geq 0 \\ &&& t - t_{\min} \geq 0 \end{aligned}$$

$$r - t \geq 0 \quad (2-13)$$

As the structural optimization is a constrained, gradient-based problem, the optimizer is working to satisfy the Karush–Kuhn–Tucker (KKT) conditions. The KKT conditions are necessarily satisfied at a structural optimum, but they do not entirely describe the suboptimization, because an optimum must also be a structural solution. Therefore, combining the structural residuals (2-12) with the KKT conditions completes the picture and allows us to fully represent the structural optimization residuals:

$$\mathbf{O} = \begin{cases} \mathbf{K}\mathbf{u} - \mathbf{f} = \mathbf{0} \equiv \mathbf{O}_S \\ \frac{dR}{dt} + \lambda_\sigma^T \frac{d\sigma}{dt} + \lambda_t^T \mathbf{I} = \mathbf{0} \equiv \mathbf{O}_L \\ \sigma_{\text{yield}} - \sigma - S_\sigma^2 = \mathbf{0} \equiv \mathbf{O}_\sigma \\ t - t_{\min} - s_t^2 = \mathbf{0} \equiv \mathbf{O}_t \\ S_\sigma \lambda_\sigma = \mathbf{0}, s_t \lambda_t = \mathbf{0} \equiv \mathbf{O}_{s\lambda} \end{cases} \quad (2-14)$$

where $\lambda^T = [\lambda_\sigma^T \quad \lambda_t^T]$ are the Lagrange multipliers for the stress and thickness constraints, respectively, and $s^T = [s_\sigma^T \quad s_t^T]$ are the slack variables associated with those constraints. The radius constraints are omitted from the KKT equations because, although they are used to guide the optimizer away from nonphysical solutions, they are not active at the optimum. Total derivatives are present in the first KKT condition, denoted as \mathbf{O}_L in the optimization residuals (15), because finding the desired sensitivities with respect to the thickness design variables requires the solution of the structural equations. This will be discussed in more detail shortly. The complete set of optimization state variables consists of four vectors, and thus $\mathbf{y}^T = [\mathbf{u}^T \quad \mathbf{t}^T \quad \mathbf{s}^T \lambda^T]$.

The multidisciplinary analysis of the asymmetric suboptimization method can be viewed as containing two separate disciplines: the aerodynamics and the structural optimization. Having identified the structural optimization residuals, \mathbf{O} , we are now able to present the corresponding coupled adjoint equations as,

$$\begin{bmatrix} \frac{\partial A}{\partial \mathbf{w}} & \frac{\partial A}{\partial \mathbf{y}} \\ \frac{\partial \mathbf{O}}{\partial \mathbf{w}} & \frac{\partial \mathbf{O}}{\partial \mathbf{y}} \end{bmatrix} \begin{bmatrix} \boldsymbol{\psi} \\ \boldsymbol{\zeta} \end{bmatrix} = - \begin{bmatrix} \frac{\partial F^T}{\partial \mathbf{w}} \\ \frac{\partial F^T}{\partial \mathbf{y}} \end{bmatrix} \quad (2-15)$$

where $\boldsymbol{\zeta}$ is the adjoint vector associated with the structural subspace optimization. The corresponding total sensitivity is,

$$\frac{dF}{d\mathbf{x}} = \frac{\partial F}{\partial \mathbf{x}} + \boldsymbol{\psi}^T \frac{\partial \mathbf{A}}{\partial \mathbf{x}} + \boldsymbol{\zeta}^T \frac{\partial \mathbf{O}}{\partial \mathbf{x}} \quad (2-16)$$

Recall that the system-level design variables (\mathbf{x}) for the proposed architecture are the jig twists of each panel ($\boldsymbol{\gamma}_{\text{jig}}$) and the sweep (Λ), whereas for the MDF method, they include the structural thicknesses (\mathbf{t}) as well.

Another valid approach to computing the systemlevel sensitivities for the new architecture is the corresponding coupled direct sensitivity equations. The direct and adjoint methods both involve the exact same partial derivative terms, but the order of operations is different. As a result, the cost of computing the sensitivities using the direct method is practically independent of the number of functions of interest, whereas the cost associated with the adjoint method is essentially independent of the number of design variables. As the new architecture removes the structural constraints from the system-level optimizer, the only function that requires total derivatives is the range equation. Thus, the use of the adjoint CPOS method is particularly advantageous in the asymmetric suboptimization method.

We will now describe the partial derivative terms of the coupled sensitivity equations, both for the MDF method and the new architecture, in more detail.

2.2.4 Aerodynamic sensitivities

The partial derivative of the aerodynamic residuals with respect to the flow variables can be decomposed as,

$$\frac{\partial \mathbf{A}}{\partial \mathbf{w}} = \begin{bmatrix} \frac{\partial b m \Lambda}{\partial \boldsymbol{\Gamma}} \\ \frac{\partial \mathbf{A}}{\partial \boldsymbol{\alpha}} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & -\mathbf{e} \\ \mathbf{e}^T & 0 \end{bmatrix} \quad (2-17)$$

which is the same matrix shown in (2-11).

The partial derivative of the aerodynamic residuals with respect to the suboptimization state variables can be broken down as follows,

$$\frac{\partial \mathbf{A}}{\partial \mathbf{y}} = \begin{bmatrix} \frac{\partial \mathbf{A}}{\partial \mathbf{u}} & \frac{\partial \mathbf{A}}{\partial \mathbf{t}} & \frac{\partial \mathbf{A}}{\partial \mathbf{s}} & \frac{\partial \mathbf{A}}{\partial \boldsymbol{\lambda}} \end{bmatrix} = \begin{bmatrix} -\frac{\partial \boldsymbol{\gamma}_{\Lambda}}{\partial \mathbf{u}} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 0 & -\frac{n}{qb^2} \frac{\partial W}{\partial \mathbf{t}} & 0 & 0 \end{bmatrix} \quad (2-18)$$

where the only direct effect of the KKT states is the dependence of the weight on the structural thicknesses. Meanwhile, the partial derivatives with respect to the structural state variables involve only the local angle of attack of each panel, which represents how

the finite-element displacements are translated into twist deflections.

We write the sensitivities with respect to the systemlevel design variables as,

$$\frac{\partial \mathbf{A}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial \mathbf{A}}{\partial \gamma_{\text{jig}}} & \frac{\partial \mathbf{A}}{\partial \Lambda} \end{bmatrix} = \begin{bmatrix} -\mathbf{I} & \frac{\partial \mathbf{A}}{\partial \Lambda} \\ \mathbf{0} & 0 \end{bmatrix} \quad (2-19)$$

where \mathbf{I} represents the identity matrix.

The partial derivatives of (2-18) and (2-19) were written specifically with the new architecture in mind. However, the MDF sensitivities involve the same terms, but rearranged in a slightly different manner. Instead of being grouped with the KKT variables, $\frac{\partial \mathbf{A}}{\partial \mathbf{u}}$ is written on its own, and $\frac{\partial \mathbf{A}}{\partial \mathbf{t}}$ is included as a design variable sensitivity. As $\frac{\partial \mathbf{A}}{\partial \mathbf{s}}$ and $\frac{\partial \mathbf{A}}{\partial \Lambda}$ are both zero, there is fundamentally no difference between the aerodynamic sensitivities required by either architecture.

2.2.5 Structural analysis sensitivities

The structural equations are included as part of the structural optimization residuals. Therefore, the following sensitivity terms are written explicitly for the MDF method and will be repeated later in the suboptimization sensitivities of the new architecture.

The matrix of sensitivities of the structural residuals with respect to the flow variables involves only the vector of external forces,

$$\frac{\partial \mathbf{S}}{\partial \mathbf{w}} = \begin{bmatrix} \frac{\partial \mathbf{S}}{\partial \Gamma} & \frac{\partial \mathbf{S}}{\partial \alpha} \end{bmatrix} = \begin{bmatrix} -\frac{\partial f}{\partial \Gamma} & \mathbf{0} \end{bmatrix} \quad (2-20)$$

The sensitivity with respect to the structural displacements is simply the stiffness matrix, i.e.,

$$\frac{\partial \mathbf{S}}{\partial \mathbf{u}} = \mathbf{K} \quad (2-21)$$

Finally, the partial derivative of the structural equations with respect to the system-level design variables is,

$$\frac{\partial \mathbf{S}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial \mathbf{S}}{\partial \mathbf{t}} & \frac{\partial \mathbf{S}}{\partial \gamma_{\text{jig}}} & \frac{\partial \mathbf{S}}{\partial \Lambda} \end{bmatrix} = \begin{bmatrix} \frac{\partial \mathbf{K}}{\partial \mathbf{t}} & \mathbf{0} & \frac{\partial \mathbf{K}}{\partial \Lambda} \mathbf{u} - \frac{\partial f}{\partial \Lambda} \end{bmatrix} \quad (2-22)$$

where we note that the jig twists do not directly affect the structure in this case.

2.2.6 Structural optimization sensitivities

As previously mentioned, the first KKT condition in (2-14) is written with total derivatives to indicate that it could require solving a set of residual equations. It needs to be

clarified that we are only referring to the structural residuals, because the suboptimization operates as a self-contained discipline. As the dependence of the aircraft range on the thickness design variables enters through the structural weight term, and because the weight is only a function of thickness, explicitly or otherwise, it follows that the total derivative of the range with respect to the thicknesses reduces to a partial derivative. Using the chain rule, this can be written as,

$$\frac{dR}{dt} = \frac{\partial R}{\partial t} + \frac{\partial R}{\partial \mathbf{u}} \frac{d\mathbf{u}}{dt} = \frac{\partial R}{\partial \mathbf{u}} \frac{d\mathbf{u}}{dt} \quad (2-23)$$

In terms of the state variables, the $\frac{d\sigma}{dt}$ term is dependent on $\mathbf{\Gamma}$ and \mathbf{t} , whereas the stress constraint itself is only a direct function of the displacements.

2.3 Results and discussion

that we developed for the new asymmetric suboptimization method were verified against the finitedifference approach. The two methods are compared in Table 1 for a trial with five jig twist and five thickness design variables, corresponding to five panels and five elements. The jig twist of the first panel was held fixed. The partial derivatives in the CPOS equations are calculated using the complex-step derivative approximation (Squire and Trapp 1998; Martins et al. 2003), which provides numerically exact values.

As shown in Table 1, the CPOS adjoint results agree with finite-differences to five digits. The CPOS results are more accurate due to subtractive cancellation errors in the finite-difference estimates. The main advantage of the CPOS method is clear: The computational time was only 3% of the time required by finite differences. Therefore, the CPOS method accurately models the response of our tightly coupled system, without having to re-converge the analysis or even once re-optimize the subspace problem, which offers significant savings over the finite-difference approach.

The reason the calculation of the partial derivative terms outweighs the adjoint solution is because the current aero-structural model is written in the scripting language Python, whereas the adjoint vectors are computed using a linear solve operation in a precompiled numerical package. Most large-scale scientific computing requires the use of a compiled language such as Fortran. When optimization is performed on these higher-fidelity systems, solving the system of equations required by the direct and adjoint methods typically becomes the dominant expense, and the use of the adjoint method becomes more

advantageous. As shown in Table 2, increasing the number of design variables from 5 to 50 does not effect the time required to compute the adjoint vectors. The only cost increase is due to the dependence of the aerodynamic residuals on the jig twists.

2.4 Conclusions

We developed a new subspace optimization method for performing aero-structural design that uses a coupled adjoint sensitivity method for determining the required system-level derivatives. We showed that this asymmetric suboptimization approach offers computational advantages when compared to the traditional MDF method: it simplifies the system-level problem, resulting in fewer calls to the MDA module, fewer total iterations, and fewer evaluations of the aerodynamic analysis.

Simple extrapolations of the results demonstrate that the new architecture becomes increasingly attractive, as either the number of aerodynamic design variables or the fidelity of the aero-structural analysis is increased. This was possible due to the sensitivity equations that were developed, which involve including the structural optimization residuals in the coupled adjoint sensitivity equations.

More generally, the adjoint CPOS method developed herein can be applied to any MDO problem and is particularly suited to problems where there are large discrepancies between the computational costs of the disciplinary solvers.