



导言

计算流体力学分析 (Computational Fluid Dynamics, CFD) 的基本定义是通过计算机进行数值计算, 模拟流体流动时的各种相关物理现象, 包含流动、热传导、声场等。计算流体力学分析广泛应用于航空航天器设计、汽车设计、生物医学工业、化工处理工业、涡轮机设计、半导体设计等诸多工程领域。

本章将介绍流体力学的基础理论、计算流体力学的基础知识和常用的CFD软件。



学习目标

- ❖ 掌握流体力学分析的基础理论
- ❖ 通过实例掌握流体力学分析的过程
- ❖ 掌握计算流体力学的基础知识
- ❖ 了解常用的CFD软件

1.1 计算流体力学基础

本节介绍计算流体力学的一些重要的基础知识, 包括计算流体力学的基本概念、求解过程、数值求解方法等。了解计算流体力学的基础知识, 将有助于理解ICEM CFD软件中相应的设置方法, 是做好工程模拟分析的根基。

1.1.1 计算流体力学的发展

计算流体力学是20世纪60年代起伴随计算科学与工程(Computational Science and Engineering, CSE)迅速崛起的一门学科分支, 经过半个世纪的迅猛发展, 这门学科已经相当成熟, 一个重要的标志是近几十年来, 各种CFD通用软件陆续出现, 成为商品化软件, 服务于传统的流体力学和流体工程领域, 如航空、航天、船舶、水利等。

由于CFD通用软件的性能日益完善,应用范围也不断扩大,在化工、冶金、建筑、环境等相关领域中被广泛应用,现在我们利用它来模拟计算平台内部的空气流动状况,也算是在较新的领域中的应用。

现代流体力学研究方法包括理论分析、数值计算和实验研究三个方面。这些方法针对不同的角度进行研究并相互补充。理论分析研究能够表述参数影响形式,为数值计算和实验研究提供有效的指导;试验是认识客观现实的有效手段,可验证理论分析和数值计算的正确性;计算流体力学通过提供模拟真实流动的经济手段补充理论和试验的空缺。

更重要的是,计算流体力学提供了廉价的模拟、设计和优化工具,并且提供了分析三维复杂流动的工具。在复杂情况下,测量往往是很困难的,甚至是不可能的,而计算流体力学则能方便地提供全部流场范围的详细信息。与试验方法相比,计算流体力学具有以下优势:参数选择灵活、成本较低且在模拟过程中不会对流场造成干扰。出于计算流体力学的这种优点,我们选择它来进行模拟计算。简单来说,计算流体力学扮演的角色是通过直观地显示计算结果对流动结构进行仔细的研究。

计算流体力学在数值研究方面大体上沿两个方向发展:一个是在简单的几何外形下,通过数值方法来发现一些基本的物理规律和现象,或者发展更好的计算方法;另一个则为解决工程实际需要,直接通过数值模拟进行预测,为工程设计提供依据。理论的预测出自数学模型的结果,而不是出自一个实际的物理模型的结果。计算流体力学是多领域交叉的学科,涉及计算机科学、流体力学、偏微分方程的数学理论、计算几何、数值分析等。这些学科的交叉融合,相互促进和支持,推动了学科的深入发展。

CFD方法是对流场的控制方程用计算数学的方法将其离散到一系列网格节点上求其离散的数值解的一种方法。控制所有流体流动的基本定律是质量守恒定律、动量守恒定律和能量守恒定律,由它们分别导出连续性方程、动量方程(N-S方程)和能量方程。应用CFD方法进行平台内部空气流场模拟计算时,首先需要选择或建立过程的基本方程和理论模型,依据的基本原理是流体力学、热力学、传热传质等平衡或守恒定律。

由基本原理出发可以建立质量、动量、能量、湍流特性等守恒方程组,如连续性方程、扩散方程等。这些方程构成非线性偏微分方程组,不能用经典的解析法,只能用数值方法求解。

求解上述方程必须先给定模型的几何形状和尺寸,确定计算区域并给出恰当的进出口、壁面及自由面的边界条件,同时需要适宜的数学模型,以及包括相应的初值在内的过程方程的完整数学描述。

求解的数值方法主要有有限差分法(Finite Difference Method, FDM)、有限元法(Finite Element Method, FEM)及有限分析法(Finite Analytic Method, FAM),应用这些方法可以将计算域离散为一系列的网格并建立离散方程组,离散方程的求解是由一组给定的猜测值出发迭代推进,直至满足收敛标准。常用的迭代方法有Gauss-Seidel迭代法、TDMA(Tri-Diagonal Matrix Algorithm)方法、SIP(Simultaneous Iterative Procedure)法及LSORC(Locally Optimized Successive Over-Relaxation with Chebyshev)法等。利用上述差分方程及求解方法即可以编写计算程序或选用现有的软件实施过程的CFD模拟。

1.1.2 计算流体力学的求解过程

CFD数值模拟一般遵循以下5个步骤:

- 步骤 01** 建立所研究问题的物理模型，在将其抽象为数学、力学模型之后，确定要分析的几何体的空间影响区域。
- 步骤 02** 建立整个几何形体及其空间影响区域，即计算区域的CAD模型，对几何体的外表面和整个计算区域进行空间网格划分。网格的稀疏和网格单元的形状都会对以后的计算产生很大影响。不同的算法格式为保证计算的稳定性和计算效率，一般对网格的要求也不一样。
- 步骤 03** 加入求解所需要的初始条件，入口与出口处的边界条件一般为速度、压力条件。
- 步骤 04** 选择适当的算法，设定具体的控制求解过程和精度的一些条件，对所需分析的问题进行求解，并保存数据文件结果。
- 步骤 05** 选择合适的后处理器（Post Processor）读取计算结果文件，分析并显示出来。

以上这些步骤构成了CFD数值模拟的全过程，其中数学模型的建立是理论研究的课堂，一般由理论工作者完成。

1.1.3 数值模拟方法和分类

在运用CFD方法对一些实际问题进行模拟时，通常需要设置工作环境、边界条件和选择算法等，特别是算法的选择，对模拟的效率及其正确性有很大影响，需要特别重视。要正确设置数值模拟的条件，有必要了解数值模拟的过程。

随着计算机技术和计算方法的发展，许多复杂的工程问题都可以采用区域离散化的数值计算并借助计算机得到满足工程要求的数值解。数值模拟技术是现代工程学形成和发展的重要动力之一。

区域离散化就是用一组有限个离散的点来代替原来连续的空间，实施过程是把所计算的区域划分成很多互不重叠的子区域，确定每个子区域的节点位置和该节点所代表的控制体积。

节点是指需要求解的未知物理量的几何位置、控制体积、应用控制方程或守恒定律的最小几何单位。一般把节点看成控制体积的代表。控制体积和子区域并不总是重合的。在区域离散化过程开始时，由一系列与坐标轴相应的直线或曲线簇划分出来的小区域称为子区域。网格是离散的基础，网格节点是离散化物理量的存储位置。

常用的离散化方法有有限差分法、有限元法和有限体积法。

1. 有限差分法

有限差分法是数值解法中最经典的方法。它是将求解区域划分为差分网格，使用有限个网格节点代替连续的求解域，然后将偏微分方程（控制方程）的导数用差商代替，推导出含有离散点上有限个未知数的差分方程组。

有限差分法的产生和发展较早，技术也相对成熟，广泛应用于求解双曲线和抛物线型问题。然而，当求解边界条件复杂，尤其是椭圆形问题时，不如有限元法或有限体积法方便。

构造差分的方法有多种，目前主要采用的是泰勒级数展开方法。其基本的差分表达式主要有4种格式，即一阶向前差分、一阶向后差分、一阶中心差分和二阶中心差分，其中前两种格式为一阶计算精度，后两种格式为二阶计算精度。通过对时间和空间的不同差分格式进行组合，可以形成多种计算方案。

2. 有限元法

有限元法是将连续的求解域划分成适当形状的许多微小单元，并在各小单元上构造插值函数，然后根据极值原理（例如变分原理或加权余量法）将问题的控制方程转换为所有单元上的有限元方程，把总体的极值作为各单元的极值之和，即将局部单元的极值相加，形成嵌入指定边界条件的代数方程组，求解该方程组，可以得到各节点上待求的函数值。这种方法对椭圆形问题有更好的适应性。

需要注意的是，有限元法的求解速度比有限差分法和有限体积法慢，在商用CFD软件中的应用并不广泛。目前常用的商用CFD软件中，只有FIDAP采用的是有限元法。

3. 有限体积法

有限体积法也称为控制体积法，是将计算区域划分为网格，并使每个网格点周围有一个互不重复的控制体积。将待解的微分方程对每个控制体积积分，从而得到一组离散方程。其中的未知数是网格节点上的因变量。有限体积法的基本思想是子域法结合离散化，其基本原理易于理解，并能得出直接的物理解释。离散方程的物理意义是因变量在有限大小的控制体积中的守恒原理，类似于微分方程表示因变量在无限小的控制体积中的守恒原理。

有限体积法得出的离散方程要求因变量的积分守恒对任意一组控制体积都得到满足，对整个计算区域自然也得到满足，这是有限体积法吸引人的优点之一。与一些离散方法（如有限差分法）相比，有限体积法即使在粗网格情况下，也显示出准确的积分守恒。

就离散方法而言，有限体积法可视作有限元法和有限差分法的中间产物，三者各有所长。有限差分法直观、理论成熟，精度可选，但处理不规则区域较为烦琐。虽然网格生成可以使有限差分法应用于不规则区域，但对于区域的连续性等要求严格。有限差分法易于编程和并行化。有限元法适合处理复杂区域，精度可选，但内存和计算量较大，且并行化不如有限差分法和有限体积法直观。有限体积法适用于流体计算，可以应用于不规则网格，适用于并行计算，但其精度基本上只能是二阶。有限元法在应力应变、高频电磁场方面的特殊优点正逐渐受到重视。

由于ANSYS CFD是基于有限体积法的，因此下面将以有限体积法为例介绍数值模拟的基础知识。

1.1.4 有限体积法的基本思想

有限体积法是从流体运动积分形式的守恒方程出发来建立离散方程的。三维对流扩散方程的守恒型微分方程如下：

$$\frac{\partial(\rho\phi)}{\partial t} + \frac{\partial(\rho u\phi)}{\partial x} + \frac{\partial(\rho v\phi)}{\partial y} + \frac{\partial(\rho w\phi)}{\partial z} = \frac{\partial}{\partial x}(K \frac{\partial\phi}{\partial x}) + \frac{\partial}{\partial y}(K \frac{\partial\phi}{\partial y}) + \frac{\partial}{\partial z}(K \frac{\partial\phi}{\partial z}) + S_\phi \quad (1-1)$$

其中， ϕ 是对流扩散物质函数，如温度、浓度。

式（1-1）用散度和梯度表示：

$$\frac{\partial}{\partial t}(\rho\phi) + \text{div}(\rho u\phi) = \text{div}(K \text{grad}\phi) + S_\phi \quad (1-2)$$

将式（1-2）在时间步长 Δt 内对控制体体积 CV 积分，可得：

$$\begin{aligned} & \int_{CV} \left(\int_t^{t+\Delta t} \frac{\partial}{\partial t} (\rho\phi) dt \right) dV + \int_t^{t+\Delta t} \left(\int_A n \cdot (\rho u \phi) dA \right) dt \\ &= \int_t^{t+\Delta t} \left(\int_A n \cdot (K \text{grad} \phi) dA \right) dt + \int_t^{t+\Delta t} \int_{CV} S_\phi dV dt \end{aligned} \quad (1-3)$$

其中，散度积分已用格林公式转换为面积积分， A 为控制体的表面积。

该方程的物理意义是： Δt 时间段和体积 CV 内 $\rho\phi$ 的变化，加上 Δt 时间段通过控制体表面的对流量 $\rho u \phi$ ，等于 Δt 时间段通过控制体表面的扩散量，加上 Δt 时间段控制体 CV 内源项的变化。

例如，一维非定常热扩散方程：

$$\rho c \frac{\partial T}{\partial t} = \frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) + S \quad (1-4)$$

在 Δt 时段和控制体积 CV 内部对式(1-4)积分：

$$\int_t^{t+\Delta t} \int_{CV} \rho c \frac{\partial T}{\partial t} dV dt = \int_t^{t+\Delta t} \int_{CV} \frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) dV dt + \int_t^{t+\Delta t} \int_{CV} S dV dt \quad (1-5)$$

如图1-1所示，式(1-5)可改写成如下形式：

$$\int_w^e \int_t^{t+\Delta t} \rho c \frac{\partial T}{\partial t} dt = \int_t^{t+\Delta t} \left[\left(k A \frac{\partial T}{\partial x} \right)_e - \left(k A \frac{\partial T}{\partial x} \right)_w \right] dt + \int_t^{t+\Delta t} \bar{S} \Delta V dt \quad (1-6)$$

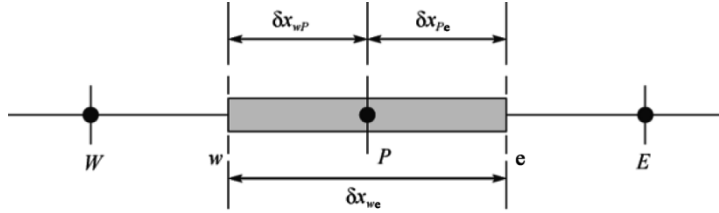


图1-1 一维有限体积单元示意图

式(1-6)中， A 是控制体面积， ΔV 是体积， $\Delta V = A \Delta x$ ， Δx 是控制体宽度， $\bar{S}$ 是控制体中的平均源强度。设 P 点 t 时刻的温度为 T_p^0 ，而 $t + \Delta t$ 时的 P 点温度为 T_p ，则式(1-6)可转换为：

$$\rho c (T_p - T_p^0) \Delta V = \int_t^{t+\Delta t} \left[k_e A \frac{T_E - T_p}{\delta x_{pE}} - k_w A \frac{T_p - T_w}{\delta x_{wp}} \right] dt + \int_t^{t+\Delta t} \bar{S} \Delta V dt \quad (1-7)$$

为了计算式(1-7)中右端的 T_p 、 T_E 和 T_w 对时间的积分，引入一个权数 $\theta = 0 \sim 1$ ，将积分表示成 t 和 $t + \Delta t$ 时刻的线性关系：

$$I_T = \int_t^{t+\Delta t} T_p dt = [\theta T_p + (1 - \theta) T_p^0] \Delta t \quad (1-8)$$

式(1-7)可改写成：

$$\rho c \left(\frac{T_p - T_p^0}{\Delta t} \right) \Delta x = \theta \left[\frac{k_e (T_E - T_p)}{\delta x_{pE}} - \frac{k_w (T_p - T_w)}{\delta x_{wp}} \right] + (1 - \theta) \left[\frac{k_e (T_E^0 - T_p^0)}{\delta x_{pE}} - \frac{k_w (T_p^0 - T_w^0)}{\delta x_{wp}} \right] + \bar{S} \Delta x \quad (1-9)$$

因为上式左端第二项中 t 时刻的温度为已知，所以该式是 $t + \Delta t$ 时刻 T_p 、 T_E 、 T_w 之间的关系式。

列出计算域上所有相邻三个节点上的方程，则可形成求解域中所有未知量的线性代数方程，给出边界条件后可求解代数方程组。

由于流体运动的基本规律都是守恒的，而有限体积法的离散形式也是守恒的，因此有限体积法在流体流动计算中应用广泛。

1.1.5 有限体积法的求解方法

控制方程被离散化后，就可以进行求解了。下面介绍几种常用的压力与速度耦合求解算法，分别是SIMPLE算法、SIMPLEC算法和PISO算法。

1. SIMPLE算法

SIMPLE算法是目前实际工程中应用最为广泛的一种流场计算方法，它属于压力修正法的一种。该方法的核心是采用“猜测-修正”的过程，在交错网格的基础上计算压力场，从而达到求解动量方程的目的。

SIMPLE算法的基本思想为：对于给定的压力场，求解离散型时的动量方程，得到速度场。因为压力是假定的或不精确的，这样得到的速度场一般不能满足连续性方程的条件，所以必须对给定的压力场进行修正。修正的原则是修正后的压力场相对应的速度场能够满足这一迭代层次上的连续方程。

根据这个原则，把由动量方程的离散形式规定的压力与速度的关系代入连续方程的离散形式，从而得到压力修正方程，再由压力修正方程得到压力修正值。然后，根据修正后的压力场，求得新的速度场。最后检查速度场是否收敛。若不收敛，则用修正后的压力值作为给定的压力场开始下一层次的计算，直到获得收敛的解为止。上述过程的核心问题在于如何获得压力修正值，以及如何根据压力修正值构造速度修正方程。

2. SIMPLEC算法

SIMPLEC算法与SIMPLE算法的思路基本一致，不同之处是SIMPLEC算法在通量修正方法上有所改进，加快了计算的收敛速度。

3. PISO算法

PISO算法的压力速度耦合格式是SIMPLE算法族的一部分，它基于压力和速度之间的高度近似关系。SIMPLE和SIMPLEC算法的一个限制是，在解出压力校对方程后，新的速度值和相应的流量可能不满足动量平衡，因此必须重复计算直到达到平衡。

为了提高计算效率，PISO算法引入了两个附加的校正步骤：相邻校正和偏斜校正。PISO算法的核心思想是消除SIMPLE和SIMPLEC算法在压力校对方程求解阶段所需的重复计算。通过一个或多个附加的PISO循环，校正后的速度将更接近满足连续性和动量方程的要求。这一迭代过程也被称为动量校正或邻近校正。

尽管PISO算法在每次迭代中可能需要更多的CPU时间，但它显著减少了达到收敛所需的迭代次数，特别是在处理瞬态问题时，这一优势尤为明显。对于具有一定倾斜度的网格，单元表面的质量流量校正和邻近单元压力校正差值之间的关系相对简单。由于沿着单元表面的压力校正梯度分量最初是未知的，因此需要进行一个类似于PISO邻近校正的迭代步骤。

在初始化压力校正方程的解之后,重新计算压力校正梯度,并用新计算出的值更新质量流量校正。这个过程称为偏斜校正,它极大地减少了在计算高度扭曲网格时遇到的收敛性问题。PISO偏斜校正使我们能够在基本相同的迭代步骤中,从高度倾斜的网格上获得与更为正交的网格相当的结果。

1.2 网格概述

CFD计算分析的第一步是生成网格,即对空间上连续的计算区域进行剖分,把它划分成多个子区域,并确定每个区域中的节点。

由于实际工程计算中大多数计算区域较为复杂,因此不规则区域内网格的生成是计算流体力学的一个十分重要的研究领域。实际上,CFD计算结果最终的精度及计算过程的效率主要取决于所生成的网格与所采用的算法。

1.2.1 网格划分技术

现有的各种网格生成方法在特定条件下都有其优势和局限性,同样,各种求解流场的算法也各有其适用范围。一个成功且高效的数值计算,只有在网格生成和流场求解算法之间实现良好的匹配时才能达成。

自从1974年Thompson等提出适体坐标生成方法以来,网格生成技术在计算流体力学和传热学领域的重要性日益被研究者所认识。

网格生成技术的核心思想是根据物理问题的特点构建合适的网格布局,并将原始物理坐标 (x,y,z) 中的基本方程转换到计算坐标 (ξ,η,ζ) 中的均匀网格上求解,以此提高计算精度。

总体而言,CFD计算中使用的网格大致可分为结构化网格和非结构化网格两大类。在数值计算中,无论是正交还是非正交曲线坐标系生成的网格,都属于结构化网格。其特点是每个节点与其邻点之间的连接关系是固定且隐含在网格生成过程中的,因此我们不需要额外的数据来确定节点与其邻点之间的联系。严格来说,结构化网格指的是网格区域内所有内部点都具有相同数量的邻接单元。结构化网格主要有以下几个优点:

- (1) 网格生成速度快。
- (2) 网格质量高。
- (3) 数据结构简单。
- (4) 对曲面或空间的拟合大多数采用参数化或样条插值的方法得到,使得区域光滑,更贴近实际模型。
- (5) 易于实现区域边界的拟合,适合流体动力学和表面应力集中等方面的计算。

结构化网格最明显的缺点是适用范围相对较窄。尤其是随着近几年计算机和数值方法的快速发展,人们对求解区域复杂性的要求越来越高,在这种情况下,结构化网格生成技术就显得力不从心了。

在结构化网格中,虽然每个节点及其控制容积的几何信息需要存储,但节点的邻接关系可以依据网格编号的规律自动得出,因此不需要专门存储这类信息,这也是结构化网格的一大优点。

当计算区域较为复杂时,即使采用网格生成技术也可能难以妥善处理不规则区域。在这种情况下,可以采用组合网格,也称为块结构化网格。这种方法将整个求解区域划分为若干小块,每个小块内部使用结构化网格。块与块之间可以是并接的,即通过一条公共边相连,也可以是部分重叠的。这种网格生成方法既保留了结构化网格的优点,又避免了需要一条网格线贯穿整个计算区域的限制,为处理不规则区域提供了便利,目前应用广泛。在这种网格生成方法中,关键环节是块与块之间的信息传递。

与结构化网格的定义相对应,非结构化网格指的是网格区域内的内部点不具有相同数量的毗邻单元,即不同内点相连的网格数目不同。从定义上可以看出,结构化网格和非结构化网格之间存在重叠,即非结构化网格中可能包含结构化网格的部分。

非结构化网格技术自20世纪60年代起开始发展,主要是为了弥补结构化网格在处理任意形状和任意连通区域网格剖分方面的不足。

由于非结构化网格的生成技术较为复杂,随着求解区域复杂性的增加,对非结构化网格生成技术的要求也在不断提高。到了20世纪90年代,非结构化网格的文献数量达到了高峰。根据目前的文献调查,非结构化网格生成技术中,平面三角形的自动生成技术已经相对成熟,而平面四边形网格的生成技术也正在逐步成熟。

1.2.2 结构化网格

结构化网格生成方法主要有两种,即单块结构网格生成和分区结构网格生成。

1. 单块结构网格生成技术

1) 代数方法

在物理平面上生成适体坐标系,即在物理平面上构建一个与求解区域边界相匹配的网格系统,使得在计算平面的直角坐标系 $\xi-\eta$ 中,与物理平面求解区域相对应的计算区域呈现为正方形或矩形。

作为网格生成的已知条件,物理平面上求解区域边界上的节点分布是预先确定的,而在计算平面上,网格通常是均匀分布的。

因此,如果将物理平面上节点的位置(以其半径 $r(x,y)$ 为代表)视为计算平面上 ξ,η 的函数,那么生成网格的过程就是已知计算平面边界上每一点的 $r(\xi,\eta)$,需要确定计算区域内每个节点相应的 $r(\xi,\eta)$ 。

显然,对边界上的已知值进行插值是获取区域内各节点值的一种简单直接的方法。这种生成网格的方法旨在寻找合适的插值函数,这种方法被称为代数法。这是因为用于生成网格的表达式都是代数方程。

接下来讨论生成网格的无限插值法,也称为无限变换。为了便于说明问题,首先以线性插值为例进行分析。如果在 ξ 、 η 两个方向上分别应用Lagrange线性插值,则有:

$$r(\xi,\eta) = \sum_{i=1}^2 h_i \left(\frac{\xi}{L} \right) r(\xi_i, \eta) \quad (1-10)$$

$$r(\xi,\eta) = \sum_{j=1}^2 h_j \left(\frac{\eta}{M} \right) r(\xi, \eta_j) \quad (1-11)$$

其中, L 、 M 分别为 ξ 方向与 η 方向的区域长度。这种两个方向的插值(双向的插值)称为无限插值,

因为它对 $\xi = 0 \sim \xi = L$ 及 $\eta = 0 \sim \eta = M$ 的整个计算范围内的空间位置进行了插值, 所以可以认为插值的点数是无限的。

现在从式 (1-10) 和式 (1-11) 得出一个无限插值的统一表达式:

$$r_{\text{TPI}}(\xi, \eta) = \sum_{j=1}^{N_j} h_j \left(\frac{\eta}{M} \right) r(\xi, \eta_j) + \sum_{i=1}^{N_i} h_i \left(\frac{\xi}{L} \right) \left[r(\xi_i, \eta) - \sum_{j=1}^{N_j} h_j \left(\frac{\eta}{M} \right) r(\xi_i, \eta_j) \right] \quad (1-12)$$

式 (1-12) 中对两条 η 为常数 ($\eta = 0$ 、 $\eta = M$) 及与之相交的两条 ξ 为常数 ($\xi = 0$ 、 $\xi = L$) 的曲线进行了拟合, 同时对位于 $0 < \eta < M$ 及 $0 < \xi < L$ 的四边形范围内的点进行了位置矢量 $r(\xi, \eta)$ 的插值, 于是完成了在 $0 \leq \eta \leq M$ 、 $0 \leq \xi \leq L$ 的矩形范围内的网格生成。

2) 保角变换方法

保角变换方法是利用解析的复变函数来完成物理平面到计算平面的映射。保角变换方法的主要优点是能精确地保证网格的正交性, 主要缺点是比较复杂的边界形状, 有时难以找到相应的映射关系式且局限于二维问题。

3) 微分方程方法

在微分方程方法中, 物理空间坐标和计算空间坐标之间是通过偏微分方程组联系起来的。

根据生成贴体网格所采用的偏微分方程的类型, 可以将其分为椭圆形方程方法、双曲型方程方法和抛物型方程方法。其中, 椭圆形方程方法最为常用, 这是因为对于大多数实际流体力学问题, 物理空间中的求解域通常是具有复杂几何形状的已知封闭边界区域, 并且在这些封闭边界上的计算坐标对应值是预先给定的。

拉普拉斯方程是最简单的椭圆形方程, 但泊松方程由于其非齐次项可以用来调节求解域中网格密度的分布, 因此被更广泛地使用。

如果只在求解域的一部分边界上规定了计算坐标值, 那么可以采用抛物型或双曲型偏微分方程来生成网格。例如, 在流场的内边界是已知的, 而外边界是任意的情况下, 就可以使用这种方法。

4) 变分原理方法

变分原理方法通过将生成网格所需满足的条件表达为某个目标函数 (泛函) 的极值问题。这种方法常用于创建自适应网格, 因为自适应网格的要求可以较为方便地通过某个变分原理来表述。随后, 可以推导出与该变分原理相对应的偏微分方程, 即欧拉方程。

2. 分区结构网格生成方法

以上是单块结构网格的生成方法, 对于复杂多部件或多体的实际工程外形, 如战斗机和捆绑火箭, 生成统一的贴体网格相当困难, 即使勉强生成, 网格质量也无法保证, 从而影响流场数值计算效果。

为了克服上述困难, CFD 工作者发展了分区网格和分区计算方法。它的基本思想是根据整体外形特点, 先将整个计算域分成若干子域, 然后在每个子域内分别生成网格并进行数值计算, 各子域间的信息传递通过边界处的耦合条件来实现。

常用的分区结构网格方法有三种, 即组合网格、搭接网格和重叠网格。

1) 组合网格

组合网格各子域之间没有重叠, 要求子域交接面上的网格节点重合。生成步骤大致为: 根据外形

和流动特点将整个计算域分区并确定每个区中的网格拓扑；在上述几何处理的基础上，按网格疏密的要求生成各部件或各体表面的网格。

注意，在此阶段应确保各子域交接面处的网格节点重合。在计算域被划分为多个子域后，相邻子域之间的公共交接面通常是一个空间曲面。这个空间曲面的位置、方向及其上的网格分布对相邻子域内空间网格的生成过程和质量有着重要影响。在生成交接面上的网格时，应根据实际情况选择适当的方法。

一旦交接面上的网格生成完成，各子域的边界也就确定了。此时，可以使用代数方法或求解偏微分方程的方法来生成各子域的空间网格。

2) 搭接网格

搭接网格的子域间同样不存在重叠区域，但与组合网格不同，它不要求子域交接面上的网格节点必须重合。在这种情况下，可以在每个子域内独立生成各自的空间网格，而不必先在子域间的交接面上生成网格。这种方法可以确保每个子域内的网格质量都很高。

然而，需要注意的是，相邻子域交接面上的网格节点数量和分布应该大致相同。如果两者相差太大，可能会导致插值误差，进而影响计算结果的准确性。

3) 重叠网格

重叠网格技术允许在整个计算域的分区过程中子域间的网格发生重叠，而不要求各子域共享边界。这样的设计显著降低了各子域在生成自身网格时的难度。区域分解技术包含两个主要方面的含义：一是涉及将整个计算域划分为若干子域；而是涉及建立子域间的信息传递机制。

1.2.3 非结构化网格

前面介绍的结构化网格生成方法表明，当用于计算相对简单的流场时，结构化网格能够构建出精细的网格，并求得精确的解。然而，面对复杂外形或流场的网格划分，结构化网格的适应性就显得不足，通常需要大量的人工干预。这样，计算机自动化的优势未能得到充分发挥，而人工构造所耗费的时间往往超过了机器的工作时间，导致人力资源的浪费。

尽管结构化网格已经发展出一系列成熟的构造技术，但随着问题的复杂性增加，每种构造技术对使用者的经验要求也随之提高，这使得许多初学者感到困惑。因此，在结构化网格生成方法难以适应的领域，研究者们发展了非结构化网格的生成方法。非结构化网格因其在几何适应性方面的优势而受到青睐，随之出现了多种非结构化网格自动生成技术，主要包括四叉树/八叉树方法、Delaunay方法和阵面推进法。

1. 四叉树（二维）/八叉树方法（三维）

四叉树/八叉树方法的核心思想是，首先用一个较粗略的矩形（二维）或立方体（三维）网格覆盖包含物体的整个计算域。然后，根据网格尺度的要求，不断细分这些矩形（立方体），即将一个矩形划分为4个子矩形（二维情况下）或一个立方体划分为八个子立方体（三维情况下）。最终，这些矩形（立方体）被划分为三角形（二维）或四面体（三维）。

例如，一个没有中间点的矩形可以被划分为两个三角形，而一个没有棱上中间点的立方体可以被划分为5个或6个四面体。对于流场边界附近被边界切割的矩形（立方体），需要考虑各种可能的情况，并进行特殊的划分。

二叉树/八叉树方法直接将矩形/立方体划分为三角形/四面体, 由于不涉及邻近点面的查找, 以及邻近单元间的相交性和相容性判断等问题, 因此网格生成速度较快。然而, 这种方法的缺点是网格质量可能较差, 特别是在流场边界附近, 被切割的矩形/立方体的形状可能非常不规则, 这可能导致由此划分的三角形/四面体的质量难以保证。

尽管存在这些局限性, 二叉树/八叉树作为一种可以提高搜索效率的数据结构, 已经被广泛应用于阵面推进法和Delaunay方法中, 以优化网格生成过程。

2. Delaunay方法

Delaunay三角化是基于Dirichlet在1850年提出的一种理论, 该理论利用一组已知点将平面划分为凸多边形。该理论的核心思想是: 如果平面上存在一组点, 那么可以将平面划分为不重叠的Dirichlet子域(也称为Voronoi子域)。每个Dirichlet子域内包含点集中的一个点, 且该子域内的任意点P到该点的距离小于到点集中的其他点的距离。将相邻Voronoi子域的边界点连接起来, 就构成了唯一的Delaunay三角网格。

CFD领域的研究者将上述Dirichlet理论简化为Delaunay准则, 即每个三角形的外接圆内不包含除其三个顶点之外的其他节点, 从而简化了三角形的划分方法。具体来说, 从一个人工构造的简单初始三角形网格系开始, 引入一个新点, 标记并删除不满足Delaunay准则的三角形单元, 形成一个多边形空洞, 然后将新点与多边形空洞的顶点连接, 构成新的Delaunay网格系。重复这一过程, 直到网格系达到预期的分布。

Delaunay方法的一个显著优点是它能够使给定集中的每个三角形单元的最小角尽可能地大, 从而得到尽可能等边的高质量三角形单元。此外, 由于Delaunay方法在插入新点时同时生成多个单元, 因此网格生成效率较高, 并且可以直接推广到三维问题。

然而, Delaunay方法也存在不足之处, 它可能在流场边界以外的区域构成非凸域或与边界相交的单元, 即不能保证流场边界的完整性。为了实现任意外形的非结构化网格生成, 需要对边界附近的操作施加某些限制, 这可能会使边界附近的网格失去Delaunay性质。另外, 对于三维复杂外形, 构造初始网格可能比较烦琐。

3. 阵面推进法

阵面推进法的基本思想是, 首先将流场边界划分为小型的阵元, 构成初始阵面。然后, 选择某一阵元, 并将其与流场中新插入的点或原阵面上已存在的点相连, 形成非结构化单元。随着新单元的生成, 会产生新的阵元, 形成新的阵面, 这一阵面不断向流场内部推进, 直到整个流场被非结构化网格完全覆盖。

阵面推进法具有其自身的优缺点。首先, 阵面推进法的初始阵面即为流场边界, 推进过程是阵面不断向流场内部收缩的过程, 因此不存在保证边界完整性的问题。其次, 阵面推进是一个局部过程, 相交性判断仅涉及局部邻近的阵元, 这减少了由于计算机截断误差而导致推进失败的可能性, 并且局部性使得执行过程可以在推进的任意中间状态重新开始。第三, 由于在流场内引入新点是随着推进过程自动完成的, 这有助于控制网格的步长分布。然而, 每推进一步, 仅生成一个单元, 因此阵面推进法的效率较二叉树/八叉树方法和Delaunay方法要低。推进效率低的另一个原因是在每一步推进过程中都涉及邻近点、邻近阵元的搜索及相交性判断。

另外，尽管阵面推进法的思想可以直接推广到三维问题，但在三维情况下，阵面的形状可能非常复杂，使得相交性判断变得更加烦琐。

1.3 常用的网格划分软件

为了完成网格的生成，过去多是用户自己编写计算程序，但由于网格生成的复杂性及计算机硬件条件的多样性，使得用户各自的应用程序往往缺乏通用性，而网格划分本身又有其鲜明的系统性和规律性，因此比较适合被制成通用的商用软件。

1.3.1 Gridgen

Gridgen能够轻松生成二维和三维的单块网格或分区多块对接的结构化网格，同时也支持生成非结构化网格，尽管非结构化网格的生成并非其主要优势。该软件用户友好，易于上手，用户可以在一到两周内学会生成复杂外形的网格。生成的网格可以直接用于Fluent、CFX、StarCD、Phonics等多款CFD软件，非常便捷。Gridgen功能强大，其生成的网格也可以直接被用户的计算程序读取（特别是当采用Plot3D格式输出时）。因此，Gridgen在CFD高级用户中拥有相当多的拥趸。

1.3.2 Gambit

Gambit作为Fluent的配套网格生成软件，主要设计用于为Fluent生成非结构化网格。由于Gambit输出的网格文件格式特殊，难以被其他计算流体动力学软件直接读取，因此它通常只在准备用于Fluent计算时使用。然而，由于Fluent拥有庞大的用户群体，Gambit也随之拥有相当数量的用户基础。

Gambit的强项在于生成非结构化网格，但它在生成适用于黏性计算的网格方面存在一定的局限性。

1.3.3 Hypermesh

Hypermesh的图形用户界面设计直观，易于学习。它支持直接导入现有的三维CAD几何模型，如ProE、UG、CATIA等，且导入过程高效，模型质量优良，这可以显著减少重复性工作，让用户能够将更多精力和时间投入分析计算中。

此外，Hypermesh还提供了一系列工具，用于优化和改进输入的几何模型。由于输入的几何模型可能存在间隙、重叠和缺陷，这些都可能妨碍高质量网格的自动划分。通过消除这些缺陷、填补孔洞以及压缩相邻曲面的边界，用户可以在模型内进行更大范围、更合理的网格划分，从而提升网格划分的速度和质量。Hypermesh还具备云图显示网格质量、单元质量跟踪检查等实用工具，便于用户及时发现并改进网格质量。

1.3.4 Tgrid

Tgrid是一款专业的完全非结构化网格生成软件，它在生成网格时不受几何结构的复杂性或尺寸的限制，特别适合用于复杂几何形状的网络生成。在网络生成过程中，用户只需提供边界网格，而无须

提供三维适体几何模型。此外，Tgrid还整合了Hexcore技术，能够在边界附近生成四面体网格，在远离边界的区域生成六面体网格，从而结合了四面体和六面体网格的优势。

1.3.5 ICEM CFD

ICEM CFD的前处理器主要包含4个核心模块：CAD几何建模处理、网格生成处理、网格优化处理以及网格输出处理。每个核心模块根据特定需求进一步细分为独立的子模块。

这些模块之间的整合度高，操作便捷，并配备了丰富的教程资源供用户参考。因此，ICEM CFD的前处理器具备一系列优势，包括强大的系统性、便捷的建模工具、友好的用户界面、多样的模块选择、清晰的网格划分策略、快速的运算能力、多样的接口支持以及便捷的学习途径，这些优点使其在众多网格划分软件中脱颖而出。

1.4 本章小结

本章首先介绍了计算流体力学的基础知识，随后阐述了网格生成的基本概念，最后介绍了几款常用的商用网格划分软件。通过学习本章内容，读者将能够掌握计算流体力学的基本概念，并了解目前市场上流行的商用网格划分软件。