

船舶水动力数值模拟方法的一些综述介绍，第二篇：方法综述并以 CFD-OHMUGA 为例子对方法和软件功能进行更为具体的介绍

（黄军涛科学网博文， 链接地址：<http://blog.sciencenet.cn/blog-3472103-1280473.html>）

船舶水动力数值模拟方法不断发展，水动力软件的也日新月异，更新很快，有很多专题的文章发表，但全面性的综述还需要更多的介绍。本人参考了一些资料，再结合自己多年的经验想做这方面的尝试，尽量多做全面性的综述介绍，但精力毕竟有限，广度上很难真真做到全面（比如没涉及网格产生的技术），深度上也很多停留在基本知识的介绍，只是希望这些工作能给一些人带来帮助，为他（她）们的科研和设计工作节省些时间，也希望抛砖引玉，见到更多的这方面的讨论和文章。已经发表了第一篇博文：计算流体力学的一些基础知识的综述介绍，这里介绍第二篇博文：方法综述并以 CFD-OHMUGA 为例子对方法和软件功能进行更为具体的介绍。

1. 应用领域

先举例说明一下水动力计算软件的一些应用领域，比如：1）船舶：阻力和动力（包括螺旋桨），操纵性，耐波性，自航，动力定位等。2）海洋结构体（固定或漂浮体）：结构体和波浪的相互作用。3）水下航行体。4）其他海洋流体力学问题：比如海洋环境和海洋学（波浪，波的破碎，波浪和风的相互作用），波浪能转换器，波浪对结构的撞击，green water, slashing, sloshing, 空化，oil skimming, VIV 等等。

2. 解决问题的方法

2.1. 方法分类

- 1)理论分析（解析解）。优点：精确直观。困难：复杂几何形状和复杂控制方程的问题不好求解，往往需要对几何形状及控制方程进行简化处理。
- 2)实验测量。主要优点：为研究和设计提供数据，为软件提供验证数据。主要困难：如何更好的降低成本，对某些周期长的实验则需要设法缩短时间。
- 3)数值模拟（往往是 CFD（计算流体力学）软件计算水动力）。主要优点：成本低，速度快，数据完备。

2.2. 船舶水动力数值模拟面临的一些挑战

- 1)高 Reynolds 数湍流问题（Reynolds : $1.0E+6 \sim 1.0E+8$ （模型船）， $1.0E+8 \sim 1.0E+10$ （实船））。我们知道在保证计算精度的前提下，尽量要减少网格数，以便加快计算的速度。但在中等网格数的前提下，靠近边界层的最大网格比（网格单元中最大边长和最短变长的比率）往往要达到 1: 10000, 甚至更大。这对计算的稳定性和精度带来很大的困难，需要选择或开发特殊的数值方法来克服困难。
- 2)复杂的船型形状。有的需要考虑船体的复杂的附件，比如舳龙骨，尾翼等。这对造出高质量的网格的带来大的困难。
- 3)水和空气运动界面的复杂的拓扑形状变化的多尺度的问题。比如包括各种尺度的波，波的破碎等。
- 4)水和空气大的密度比和表面张力给计算带来奇性，保持水和空气的物性阶跃特性（sharp interface）对计算方法提出高的要求。
- 5)船体和海洋漂浮体复杂的 6 个自由度的运动，多船体的计算，以及浮体位置和运动姿态的控制。

6)复杂物理现象等。

2.3. 数值模拟方法（软件）的一些分类（不考虑结构变形的流固作用问题的计算）

1) 势流方法。缺点：忽略了流体的粘性，有时候比如求船的阻力往往误差较大，另外处理有拓扑形状复杂变化的问题比如波的破碎也有困难。优点：比如 panel 方法，可以把体积问题转化为曲面的问题，加快了计算速度(当然系数矩阵是满矩阵，会在一定程度上拖累计算速度)。对势流方法，本人没有做过太多的研究课题，以后忽略不表。

2) 粘性流方法（以后下文的 CFD 在这里就专指这种方法）。

(1) 自由面问题（free surface flow: 忽略空气效应的单相流），或流体运动界面问题（moving interface flow: 水和空气的耦合的两项流问题）。船舶水动力软件一般就是这类问题。

(a) Surface track (界面追踪法, Lagrangian method): Front Tracking, Particle Based 等等。

(b) Surface capture (界面捕获法, Eulerian method)。

(c) Hybrid Lagrangian and Eulerian method. (在这不展开讲)。

(2) 非自由面或流体运动界面的问题。比如一般的螺旋桨问题的计算。

(3) 其他的涉及复杂物理现象的问题比如空化的求解方法在此省略不表。

3) 自由面或流体运动界面问题的 Surface capture 方法（只是方法举例，不做全面介绍）

(1) 结构网格（精度高）。

(a) 直角坐标的方法。比如：Sussman (VOF- Level set), USA, Hu (CIP, semi-Lagrangian scheme), Japan, CFDSHIP-Iowa V6 (VOF-Level set) USA, 等等。

(b) 曲线坐标的方法（适体网格）。比如：Cura Hochbaum, Vogt (Level set), German. CFDSHIP-Iowa V4, V4.5 (非正交曲线网格, Level Set 方法) USA. INSEAN (Level Set, single-phase), Italy. TRANSOM (DRDC), Canada. CFDSHIP-Iowa V6 (VOF-Level Set, 也包含曲线坐标), USA.

(2) 非结构网格的方法（复杂形状）

(a) Level set 方法（非质量守恒型方法，但光滑）。

比如：SURF (artificial compressible), Japan. U²NCLE (artificial compressible), USA. CFD-OHMUGA, Canada. 等等。

(b) VOF 方法（质量守恒型方法，商业软件一般为 Algebra VOF 方法, compressive method）。比如：FLUENT, CFX, STAR-CCM+ (HRIC 方法处理界面), OPENFOAM (Multi-methods, hybrid, 各种衍生的版本), European. U²NCLE (two-phase flow, density diffusion), USA. ISIS-CFD (BICS, BRICS). EMN, France. 等等，还有很多。

(c) CLSVOF (Coupled Level Set and VOF)。

3. 水动力软件需要做到的一些目标和功能

从海洋和船舶研究和设计领域的应用角度，希望能够以软件为工具帮助计算船舶的阻力和动力，操纵性，耐波性，自航，动力定位，船型设计优化，以及海洋结构体和波浪的相互作用，等等。

4. CFD-OHMUGA 为例子，介绍水动力数值模拟的全过程

4.1. CFD-OHMUGA 的目标和功能

1) 目标

开发和应用新的 CFD（计算流体力学）模型，世界领先的数值方法和软件，实现快速，稳定，和精确计算，用于复杂，挑战性和新的海洋水动力学的数值模拟计算（船舶，海洋结构等），为科学研究，工程设计，和船舶操作，提供可靠的工具。

2) 主要功能

(1) 船舶水动力计算方面的功能：MPI 并行计算海洋水动力学，比如：船舶的阻力和动力，船舶操纵性，耐波；实现螺旋桨和舵控制下自航及多船体的相互作用；另外从 CFD 功能和 3-DOF 控制器的角度看，原则上可以做动力定位，但只是缺少动力定位自动控制部分的参考资料（正在做）。另外，还可以做船型优化（结合 Optimization-OHMUGA）。

(2) 计算流体力学本身来说的功能：流体（或包括波浪）和具有复杂形状及相互独立运动的多个物体（浮体或潜体）及其独立运动的多个附件（螺旋桨，舵等控制器），在流固相互作用下的流体流动，流体自由面运动，各物体所受力，力矩，以及六个自由度的运动（姿态和轨迹）及其控制(控制器相对母体的三个旋转运动自由度)，的全过程耦合并行计算。

(3) 一些特色功能

(a)CFD-OHMUGA 和 Overset-OHMUGA 通过并行计算管道相耦合，利用 overset 网格技术专门用于解决一些复杂的问题：比如物体的相对运动，复杂物体形状，局部网格加密，等。

(b)物体 1-6 自由度运动：完全 6DOF 自由，或部分自由和约束的混合运动，可以进行计算预测（自由运动）或人为规定（约束运动）。控制器：1-3 自由度旋转运动。

(c)控制器类型：(1)附件执行器(螺旋桨，舵等实物)，(2)模型控制器（数学模型螺旋桨）。

控制目标：航速，前进运动方向，航点等。控制方法：开环，或反馈控制(PID)。

(d)线性规则和非规则入射波。

(e)多目标多约束的优化和代理模型的方法。

4.2. OHMUGA 软件功能块的结构

为了实现以上目标和功能，需要对软件的基本框架进行设计。下面以 OHMUGA 软件系列为例来进行介绍。(1) 以 CFD-OHMUGA（流体力学求解器）为核心进行水动力的计算。CFD-OHMUGA 主要采用了流体力学的控制方程和刚体动力学方程（这里不考虑结构变形）建立模型，通过采用和开发世界前沿的数值方法，计算刚体和流体的相互作用问题。(2) Overset-OHMUGA（动态 overset 网格求解器）主要是解决因为多个物体的相对运动（比如船体，螺旋桨，舵有不同的相对运动）造成的计算困难，也可用于局部加密网格，以及处理复杂的物体形状（比如带附件）的问题。Overset-OHMUGA 往往和 CFD-OHMUGA 耦合在一起处理复杂的问题。(3) Optimization-OHMUGA solver（船型优化求解器）则采用 CFD-OHMUGA 的计算结果（比如阻力），对船型进行优化。下面的总体策略图可以帮助理解以上的问题。

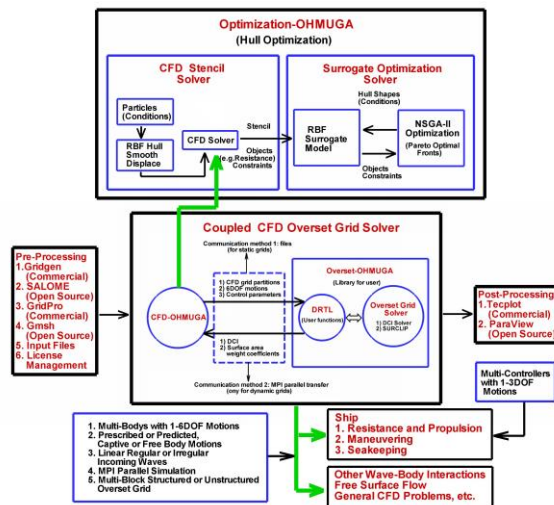


图 1 OHMUGA 软件功能块结构

4.3. 数学模型（详见 CFD-OHMUGA 用户手册）

1) 惯性坐标系(Inertia Coordinate Systems) 和非惯性坐标系(Non-inertia Coordinate Systems) 流体力学控制方程组都在惯性坐标系（地球系统(earth system)或相对地球的某个匀速直线运动的系统）下求解，而刚体动力学方程则在非惯性坐标系（船体坐标系, body-fixed system）中求解。

2) 不可压单相粘性流模型

忽略空气效应，只考虑水单相流体的流动。N-S 方程为控制方程，包括质量守恒方程（对不可压流体来说则为速度的散度为 0）和动量方程。对方程进行无量纲化处理。

3) 湍流模型

这里选用了 RANS（或 URANS: Unsteady Reynolds Averaged Navier-Stokes Equations）或 DES（Detached-Eddy Simulation）模型。采用了 Menter (AIAA Journal 1994; 32:1598–1605)的 k-omega/k-epsilon 混合模型，包括 BSL（Baseline Model）和 SST（The Shear-Stress Transport Model）。RANS 模型往往在流动的分离区耗散太大，如果想对涡结构有更好的解析，一般采用 LES 模型，但 LES 在近固体壁面处计算还是很耗时，RANS 和 LES 优点互补的混合模型 DES 模型则是一个好的选择，在近壁面处采用 RANS，远处的流体分离区则采用 LES 模型。目前的 DES 模型是在前面介绍的 Menter 的湍流模型基础上，对 k 方程的耗散项进行修改而得的（Travin et al. 2004, Advances in LES of Complex Flows, 2004, 239–254）。

4) 流体自由面捕捉方法

采用 Level set 方法（保持界面的充分光滑性）对自由面进行捕捉，包括 Level set 传输方程和保持距离函数的重新初始化方程。各变量（速度，压力，湍流变量）在自由面处的边界条件是根据水和空气界面的阶跃条件，再忽略空气效应推导出来的（Huang, Int. J. Numer. Meth. Fluids 2007; 55(8): 867-897; Paterson EG, 2003, IHR Report）。

5) 刚体动力学方程

忽略船体的结构变形，认为船体是个刚体，具有 1-6DOF（可以部分约束或完全自由）的运动，其运动满足刚体动力学方程。刚体动力学方程是固定在船体的非惯性坐标系下表示的，包括动量方程（三个线速度）和角动量方程（三个角速度）。方程中的转动惯量在这里可以不设为独立的变量，而由惯性半径来表示。当把固定在船体的坐标原点与质点重合的时候，方程的形式最简单。

这里需要指出，目前的设计是船体的控制器（螺旋桨，舵）可有 1-3DOF（可以部分约束或完全自由）的强制运动。

6) 动态重叠网格和 ALE 方法（Arbitrary Lagrangian Eulerian method, 欧拉和拉格朗日混合法）动态重叠网格相对流体可能产生不稳定的运动，动态重叠网格之间也往往有不一致的相对运动，这对 Euler 坐标系下解决问题造成困难。ALE 方法的引入是为了解决这种困难，但 ALE 方法使得各控制方程的形式发生变化，在对流项中需要用流体和网格的相对速度来代替对流速度（详见 Huang, 2020, ISOPE paper, Vol.3, pp. 2117-2124）。

7) 线性规则和非规则的入射波。

为了研究浮体和波的相互作用，需要设定入射波的边界条件和初始条件。入射波一般为分为规则和非规则两类。这里考虑的线性入射波的情况。根据线性波理论（有限水深情况则考虑了色散关系），可以得出流场的自由面，速度和压力的解析式，以此作为边界或初始条件（详见 Huang, 2020, ISOPE paper, Vol.3, pp. 2117-2124）。另外需要提出的是，CFD-OHMUGA 可以利用 Overset Grid 通过比如边界的活塞运动造波。

8) 体积力螺旋桨模型（Hough and Ordway(1964)）

鉴于实物螺旋桨的计算过程比较耗时，可以用螺旋桨模型近似处理。这里用 Hough and Ordway 的体积力模型（单向作用模型），螺旋桨提供推力和扭矩使船体产生运动，并且提供

体积力的分布（轴向和切向力呈对称分布）影响流场的分布。一条船上可以配置单个或多个螺旋桨模型。

9) 弹簧模型

这里提供一种简单的弹簧模型用于浮体的系泊。可以是单个或多个弹簧。输入条件为弹簧两端的初始位置和刚度系数。

10) 控制器

目前在 CFD-OHMUGA 中，已经添加了船体的控制器（螺旋桨，舵）模型，这种类型的控制器是主动的，可有 1-3DOF（可以部分约束或完全自由）的强制旋转运动，每条船可以添加一个或多个不同类型的控制器。控制器提供了一种灵活的方式来控制船体运动（比如 ship maneuver），复制实验条件，并通过数值模拟分析不同情况下的浮体性能（如船舶操纵性、推进加速、自航下的耐波性（航向保持或航点航行））。船体附件（如舵、实际螺旋桨）或物理模型（速度递增推进、模型螺旋桨）都可以用作控制器。CFD-OHMUGA 中实现了三种控制器，1）描述控制器，2）递增控制器，3）PID 控制器。

对于描述控制器，执行器是按照给定的函数随时间演化来运行的，这些函数通常用于模拟船舶操纵性，如回转试验、Z 形试验或其他特殊用途。递增控制器是用来逐步和顺利地加快船舶速度，舵角或螺旋桨转速，通过特定的多项式函数随时间演化。目前，在软件中设置了两个选项：线性函数和用户定义的多项式函数（最多三阶）。PID 控制器以误差为基础，通过选择与误差（P）、误差积分（I）和误差导数（D）成比例的适当系数来实现。船舶自航通常采用 PID 控制器，通过控制航向和航速达到目标值。

控制器是基于开/关信号（例如时间周期），限制动作参数和最大变化率。限制器通常是执行器的物理下限或上限，最大速率用于控制变化率。

在当前设置中，每个控制器独立于其他控制器，并且可以装配在特定的浮体（船）中以供特定用途。这意味着某些类型的控制器可以为一个浮体协同工作（例如 PID 舵和 PID 螺旋桨为船舶自航协同工作），多浮体（多船舶）问题需要不同的控制器组来完成。

控制器通常用于船舶的三种用途的研究：推进，操纵性和耐波性。采用递增或 PID 控制器，通过模型螺旋桨或真实螺旋桨可以控制船体推进（船舶前进速度）。对于操纵性（如转弯试验、Z 形试验），通常由舵（一个或多个舵）通过描述控制器来控制航向，由螺旋桨（模型或真实）通过描述或 PID 控制器来控制目标速度。耐波性（波浪中的自航，固定航向的 course-keeping 或可变航向的 waypoint）通常由舵（航向）和真实或模型螺旋桨（推进）共同控制。动力定位的问题的模拟理论上已经具备了条件，但只是缺少动力定位自动控制部分的参考资料（正在做）。

4.4. 数值方法（详见 CFD-OHMUGA 用户手册）

4.4.1. 网格

CFD-OHMUGA 采用非结构网格（如果是多块结构网格则自动转化成非结构网格的形式）的方法，其网格单元可以包括四种不同类型：四面体，六面体，三棱柱体和棱锥体（金字塔），可以是单纯的某种网格，也可以是混合网格的形式。一般六面体和三棱柱体型网格设置在边界层中，直角坐标网格一般用作背景网格或在焦点区（overset grid）以提高计算的精度。非结构网格的优点是能处理比较复杂几何形状的问题，在现实的工程问题的计算中得到大量的应用。很明显结构网格本质上是非结构网格的一种特殊情况而已。准对结构网格可以采用精致的高精度对流格式的优点，在 CFD-OHMUGA 专门对结构网格的特殊情况进行处理，设立选项开关，可以在当地网格节点处沿着曲线坐标找到邻居的关系，从而兼容对流格式的精致的高精度离散的优势。

CFD-OHMUGA还可以和Overset-OHMUGA相结合，处理相对运动的问题（船，桨和舵有不同

的运动，还有多船体之间也有不同的运动)，复杂的几何形状的问题（比如舳龙骨），以及需要局部加密网格以提高计算精度的问题。Overset-OHMUGA是专门的重叠网格的求解器（分为静态和动态两种类型），可以为CFD求解器提供DCI（Domain Connectivity Information）和面积积系数（Surface Area Weight Coefficients）。对于静态类型的重叠网格，CFD和Overset grid Solver是通过硬盘文件的读写进行信息的传递，而对于动态类型的重叠网格，CFD和Overset grid Solver是通过DRTL的函数库进行调用，在MPI并行计算的进程之间直接进行数据传递，CFD求解器和Overset网格求解器耦合的策略见图2所示。注意Overset grid Solver比CFD Solver求解速度快很多，在这里允许CFD和Overset grid Solver求解器对相同的Overset串网格可以有不同的子区域（对应并行计算的进程控制的区域）的剖分数量，这样可以节省进程的数量，在有限资源的情况下，可以为CFD安排更多的进程数。

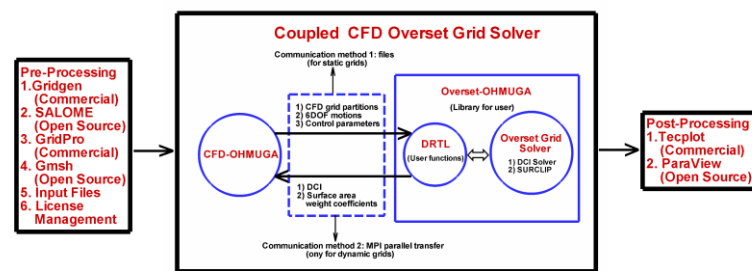


图2 Strategy of CFD and Overset grid Solver

Overset Grid Solver是处理非结构网格（网格类型和CFD-OHMUGA一样）的求解器，Overset-OHMUGA 创造性地开发了一种崭新IBA(Iterative Band Algorithm)的挖洞的方法，用迭代更新带的方法依次挖洞，迭代的次数由附件网格（比如舳龙骨，必须和目标体相连）粘贴在目标体（船体）基础网格的最大层数决定（比如舳龙骨网格粘贴在船体网格为2层，如舳龙骨网格上面再贴别的小附件的网格，则为3层）。如果没有附件网格，即便是再多的网格数量或重叠层次，1次迭代即可完成。这种方法的特点是：自动处理，水密性的条带，各网格不强制要求设等级，不用辅助的几何图形，边界附近单元不需要被保护和特别处理，允许多层Fringe插值点（可以设任何层数，只要重叠足够），多层的附件，高效的数据搜索（MPI并行及数据结构处理），孤立节点的特殊处理。如果想更详细的了解Overset-OHMUGA，可以参阅OHMUGA发表的文章和文件（Huang, 2016, ISOPE conference, Greece, June 26-July 1, Vol.3, pp. 368-375; OHMUGA用户手册）。

4.4.2. 控制方程的离散方法

1) 有限体积法

采用同位网格和有限体积法对以单元顶点为中心的median-dual方案离散。其中流体控制方程（包括湍流）采用守恒型的积分方程后再离散，Level Set方程则可选守恒或非守恒的形式。

2) 时间和空间的离散

对所有的控制方程（包括 Level Set 传输方程），在时间上采用二阶精度的 Euler 方法隐格式离散，可以采用较大的时间步长。空间上对流项采用 biased scheme 基础上二阶精度的上风格式，可有各种选项，比如一般的上风格式和带限制函数（limiter function）的方法。限制函数有 Roe's minmod (1970), Roe's superbee (1985), van Albada (1977), Van Leer (1974)等，还有专门用于非结构网格的 Bath and Jespersion (1989)和 Venkatakrisnan (1993, 1995)。

3) 速度和压力耦合方法

目前采用 PISO（Pressure-Implicit with Splitting of Operators）和投影（Projection）两种方法。

PISO 算法耗时要长很多，但质量守恒方程的残差要小些。投影法则速度相对较快。

4) Poisson 方程的离散

无论采用 PISO 或 Projection 方法，都需要求解压力 Poisson 方程。先把 Poisson 方程写成积分的形式，再采用等参元的方法构造单元内的压力插值函数，写出控制体面上的压力梯度，即可对 Poisson 方程完成离散。如果采用这种方式进行离散，对于结构网格的特殊形式是 27 点格式。对于结构网格的特殊形式，还可选用更精致的 19 点格式作为选项（详细的方法见 Huang, *Int. J. Numer. Meth. Fluids* 2007; 55(8): 867-897）。

5) Level Set 的离散方法

对 Level Set 传输方程或重新初始化方程，采用窄带技术。这种方法是把计算点只集中在界面附近的几排网格点（比如 10 排点），对于别的点则用带符号（水为正数，空气为负数）的常数直接赋值，不用再进行繁琐的计算。这种方法可以大大提高计算效率和稳定性。

此外，开创性的开发了一种几何和数据结构(KD-Tree)的方法处理 Level Set 重新初始化。用 MPI 并行化的 KD-Tree 的方法进行高效的搜索，再用几何的方法可以计算 Close point 上的 Level Set 的值，最后以此为边界值求解微分方程。这种方法再结合窄带技术可以大大提高 Level Set 重新初始化的计算速度。除此以外，在获得 Close point 值后，可以用 MPI 并行化的 Fast Marching 技术获得更为快速的重新初始化的处理，然而这种方法在处理网格尺寸差距很大的 Overset 网格时，在部分重叠区光滑性上有待提高。

6) Overset 网格的一些特殊处理

对于高精度的对流离散格式，需要更宽的模板来支持。当一些模板值刚好落在 Hole point 时，而在这些 Hole point 对应的变量值又没有很好的赋值的时候就会出现非物理的东西，使计算发散或解偏离物理实际。所以需要对这些 Hole point 进行赋值，这里是通过求解扩散方程（Laplace equation）的方法对 Hole point 进行处理。另外，在当前时间层的 Active point 如果在上个时间层是 Hole point 的话，求时间导数就会出问题。为了解决这个问题，CFD-OHMUGA 求解器会要求 Overset-OHMUGA 求解器对这些点求上个时间层的插值。

7) 刚体动力学方程的离散和求解

刚体动力学方程求解要同时用到惯性坐标系（一般为地球）和非惯性坐标系（船体），设立 2 套 6DOF 的变量，其中在惯性坐标系中 6DOF 用 3 个线性位移和 3 个旋转角 Euler 角（包括对应的速度）来表示。在非惯性坐标系下则用另外一套 6DOF 对应的变量来表示。求出流体力学方程后，把力和力矩转化到船体坐标系下的对应量。在船体坐标系下，采用时间上二阶精度的隐格式对刚体动量和角动量方程组迭代求解，求出速度的值，然后再把速度转化到惯性坐标系，最后求出线性位移和 Euler 角的值。

4.4.3. 线性方程组的并行求解

这里采用的方程主要是两类，（1）动量方程，湍流，以及 Level Set 方程，（2）压力 Poisson 方程。对于主要是抛物型方程的第（1）类问题采用 Gauss-Seidel 方法求解，如果有大量的结构网格，可以局部用 ADI 快速求解。鉴于采用了 MPI 并行计算，采用进程所属的网格的交界面处部分重叠的 Schwarz formulation 的方法作为 DD（Domain Decomposition）方法。对于椭圆型的压力方程，这里利用了开放的 PETSc 工具包（Copyrighted by UChicago Argonne, 2-clause BSD）来求解。有若干种 Krylov 子空间迭代法可以选择，常用的为 GMRES 方法和 BiCGSTAB 方法，为了获得较快的收敛速度，要选择合适的预处理方法，比如块: Block Jacobi, Incomplete LU, Incomplete Cholesky, SOR, Multi-Gridden 等。

4.4.4. 其他的方法

对于 MPI 并行计算，采用 Load-balance 的方法对原始网格进行分配，使得分割后的网格的


```

&routine_input          ! 计算路线输入参数
  it_time_srt           = 1          ! 从第一步时间层开始计算, 与 mode=0 相对应
  it_time_end           = 3000       ! 第 3000 步时间层结束
  niter_onetime         = 5          ! 每个时间层速度和压力最大的外迭代次数
  niter_firsttime       = 6          ! 第一步时间层速度和压力最大的外迭代次数
  it_save_print         = 1000       ! 每 1000 步输出计算结果
  it_save_rst           = 1000       ! 每 1000 步储存数据 (为预防停电事故,
                                     作为下一步的输入数据)
  it_save_resid         = 1          ! 每时间步输出收敛指标数据
/end

&flow_input             ! 流体流动输入参数
  reynolds_num          = 4.85d+6    ! 雷诺数=4.85d+6 (都是无量纲化的方程)
  froude_num            = 0.28       ! Froude 数=0.28
  del_t                 = 0.01d0     ! 时间步长=0.01 (0.01d0 是双精度格式,
                                     即便输入 0.01, 软件也会自动转化成双精度。)
  uinfull               = 1.d0       ! X 向来流速度=1.0, 参考 i_earth_system= 0 的注解。
  vinfull               = 0.d0       ! Y 向来流速度=0.0
  winfull               = 0.d0       ! Z 向来流速度=0.0
/end

&turbulence_input      ! 湍流输入参数
  flag_lami_turb        = 1          ! =0 层流, =1 湍流 (应用湍流模型)
  wt_bslsst             = 1.d0       ! wt_bslsst=0.0 (BSL 模型), 1.0 (SST 模型)
  i_des_mask            = 0          ! =0 RANS 模型, =1 DES 模型
/end

&freesurface_input     ! 自由面模型
  flag_freesurface     = 1          ! 1: 有自由面流动, 0: 无自由面
/end

&energy_input          ! 这部分还是空闲 (以后加各种能量模型)
/end

&usereq_input          ! 用户可以加 n 个他们自己建立的非稳态对流和扩散型的方程
/end

&immersedby_input     ! 这部分还是空闲 (以后准对 Immersed Boundary 方法)
/end

&waves_input           ! 入射波参数输入
  i_wave_type           = 1          ! 1: 规则线性波
  wave_length           = 1.5d0     ! 波长
  wave_amplit           = 5.968d-3  ! 波幅

```

/end

&controller_input ! 船舶自航各种控制器参数输入（这个例子不用设）
/end

&propeller_input ! 模型螺旋桨参数的设置（这个例子不用设）
/end

&mooring_input ! 弹簧系泊模型参数的设置（这个例子不用设）
/end

¶llel_input ! MPI 并行计算参数设置（有缺省值，什么都不设也无妨）
m_part_lays = 3 ! 拓展 3 层（可设任意数，只要邻居网格足够大）虚网格，
以便各进程之间的信息交换。
m_part_lays = 3 可满足 2 阶精度的对流格式。
/end

&solver_input ! 迭代求解参数的设置（有缺省值，什么都不设也无妨）
arfu = 0.5 ! u 变量的松弛因子（每时间层外迭代过程中）
arfv = 0.5 ! v 变量的松弛因子
arfw = 0.5 ! w 变量的松弛因子
arfp = 0.4 ! p 变量的松弛因子
arfte = 0.4 ! k 变量的松弛因子
arfed = 0.4 ! Epsilon 变量的松弛因子
arfls = 0.4 ! Level set 变量的松弛因子
res_uvw_incrit = 1.0d-05 ! 动量方程收敛标准
/end

&grids_input ! 网格参数的输入
num_cluster_os = 2 ! 一共是 2 套独立的 overset 网格
f_namei_os = hullcs bgcs ! 网格的名字（注意下面的参数是列对齐）
i_switch_us_os = 1 1 ! 网格的类型（1: 结构网格, 2: 非结构网格）
parent_ads_os = 1 1 ! 网格属于第几条船（这个例子只有一条船）
kid_ads_os = 0 0 ! 网格属于哪个控制器（这个例子没有控制器）
nmesh_split_os = 6 6 ! 网格剖分成子区域的数量（MPI 并行计算需要）
MPI 并行计算的进程数 6+6=12
/end

&motion_input ! 船体 6DOF 和控制器 3DOF 参数的设置
mt_time_order = 2 ! 刚体动力学方程离散在时间上 2 阶精度
time_start_mt = 0.d0 ! 船体从时间 t=0 开始有运动

高精度对流离散格式模板的大小。

```
/end
```

```
&routine_input
```

```
/end
```

```
&ovst_input
```

```
/end
```

```
&parallel_input
```

```
/end
```

```
&grids_input
```

```
  num_cluster_os = 2 ! 2套 Overset 网格，这个数如果不设也无妨，  
                    也会被 cfd_input.nml 中同名变量覆盖  
  f_namei_os     = hullcs bgcs ! 网格名，这个数如果不设也无妨，  
                    也会被 cfd_input.nml 中同名覆盖  
  nmesh_split_os = 2      1    ! 网格剖分数（或 MPI 进程数）这个必须设，  
                    建议设的比 cfd_input.nml 的值小，  
                    因为 Overset 求解器远远比 CFD 求解器速度快。  
  ist_out_frg_os = 1      0    ! 1: 强制设置非 active 的区域，如果发现  
                    远离船体的区域网格质量较差（比如扁长），  
                    可以用这个方法，强制定义为 1，可以人为屏蔽  
                    那些差的网格。不过设 0 也可以，软件会自动处理。  
  dist_out_frg_os = 0.3d0 0.d0 ! hullcs 网格中到船体表面距离>0.3 的所  
                    有点为非 active 点。
```

```
/end
```

```
&domain_input
```

```
/end
```

```
&dos_input
```

```
/end
```

```
&post_input
```

```
/end
```

3) 软件运行

Linux 机器中执行 MPI 运行命令运行软件：`mpiexec -np 15 ./cfdus -num_procs_dos 3`

上面命令中 `cfdus` 为 CFD 求解器执行文件，其中 `Overset-OHMUGA` 已经以库的形式包含在执行文件中。数字 3 为 `Overset-OHMUGA` 所用的进程数，15 为 `CFD-OHMUGA(12)` 和 `Overset-OHMUGA(3)` 所用的进程数的总和。

4) 后处理（数据处理）

软件输出数据文件后，用户可以通过后处理进行数据的分析。根据计算结果，下面举例给出一些图和动画。

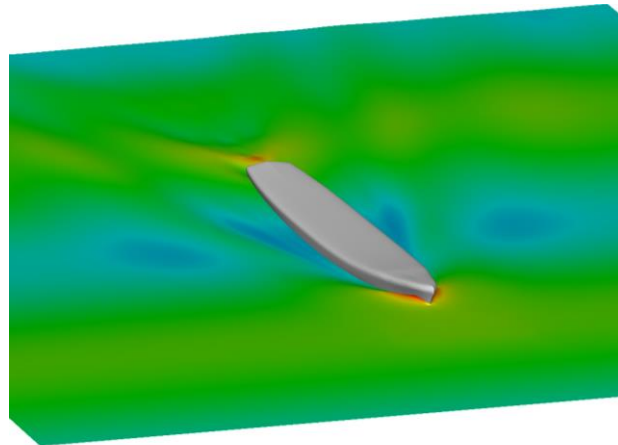


图 3 某一时刻，船的姿态和周围自由面的分布

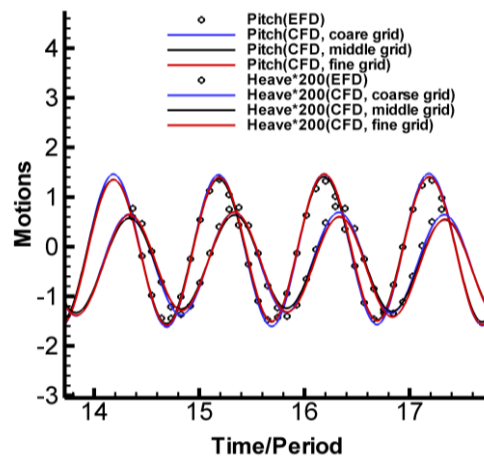


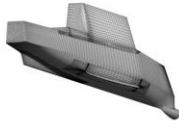
图 4 船体运动随时间的演变

船的姿态和自由面随时间的变化的详细过程，可见动画。

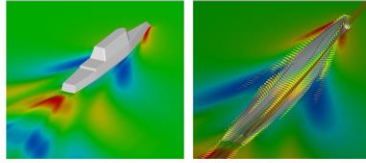
如果想看到其他更多的例子，可以参照下面的图

CFD-OHMUGA

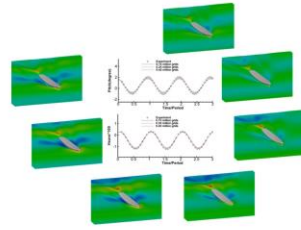
Website: ohmuga.com
Email: jhuang@ohmuga.com



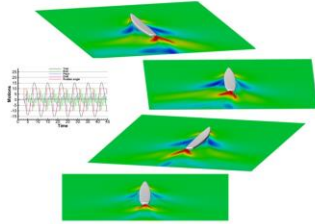
Dynamic Overset Grids
(Overset-OHMUGA)



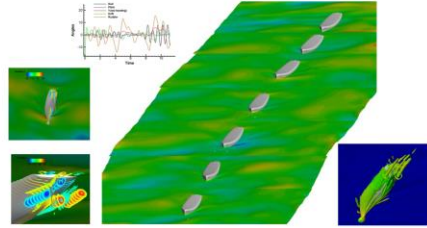
Resistance
Hull Optimization (Optimization-OHMUGA)



Pitch and Heave



Maneuvering



Course Keeping

也可点击链接 [movie](#) 观看动画介绍。