# 适用于任意网格的大规模并行 CFD 计算框架 PHengLEI

赵 钟 张来平 何 磊 何先耀 郭永恒 徐庆新

(中国空气动力研究与发展中心计算空气动力研究所 四川 绵阳 621000)

摘 要 针对计算流体力学(Computational Fluid Dynamics,CFD)大规模并行计算的需要,我们设计了适用于任意 网格类型的并行计算框架风雷(PHengLEI)软件,同时支持结构网格、非结构网格和重叠网格.为了实现并行通信与 网络类型无关且与物理求解器开发者隔离,我们设计了通用的通信模式和一种基于网格块的"三合一"的数据交换模式,即将多块结构网格块、并行分区网格块和结构/非结构网格块间的交界面数据完全标准化为数据底层后,在数据 底层统一进行交换,使得数据交换不依赖于网格类型和数据类型,从而实现任意网格类型并行通信模式的统一.作 为气动数据生产的"数值风洞",要求代码将并行隔离至底层,为此针对主流计算机系统的特点,设计了粗粒度 MPI/OpenMP 混合并行模式,领域专家只需关注求解器开发而无需关注并行通信,在提高可移植性、可扩展性的同 时尽可能兼顾封装性.针对大规模并行计算的实际工程需求,设计了并行分区、大规模并行文件存储模式,以缩短 前置处理和文件 I/O 时间.最后,分别基于结构网格、非结构网格和重叠网格,给出了工程应用领域的标模算例,以 说明 PHengLEI 并行计算框架对不同网格类型的适应性.大规模网格测试结果表明,国产定制集群上 16.38 万核 并行效率达 83%以上,"天河二号"上近 10 万核并行效率达 95%以上,展现了较好的可扩展性和并行计算效率.

关键词 并行计算; MPI/OpenMP 混合并行; CFD 并行计算框架; PHengLEI;风雷软件; HyperFLOW
 中图法分类号 TP311 DOI 号 10.11897/SP. J. 1016. 2019.02368

# PHengLEI: A Large Scale Parallel CFD Framework for Arbitrary Grids

ZHAO Zhong ZHANG Lai-Ping HE Lei HE Xian-Yao GUO Yong-Heng XU Qing-Xin (Computational Aerodynamics Institute, China Aerodynamics Research and Development Center, Mianyang, Sichuan 621000)

**Abstract** With the development and maturity of Computational Fluid Dynamics (CFD) computational methods, from simple Euler solver to RANS, RANS/LES, LES, DNS models, the demand for the amount of calculation is becoming more and more. The development of CFD heavily relies on the High Performance Computer (HPC) since the last several decades. Although several HPC systems have been built to meet the demand of science research, such as SUMMIT in USA and Sunway TaihuLight, TianHe-2 in China, the existing CFD codes are difficult to adapt the HPC hardware architectures on a unified framework. The most key factor is the existing software is based on only structured or unstructured computational mesh and using completely different parallel computational architecture. In addition, each physical solver developer is required to possess parallel programming ability, rather than isolating the parallel to the bottom layers, which would result in substantial coding difficulty. To overcome this pain point, a parallel CFD computational framework, named PHengLEI, for arbitrary grids has been designed to meet

收稿日期:2017-12-01;在线出版日期:2018-10-09.本课题得到国家重点研发计划(2016YFB020071)和国家自然科学基金(11532016, 91530325)资助. 赵 钟,博士,助理研究员,中国计算机学会(CCF)会员,主要研究方向为非结构网格生成、计算流体力学及网格生成软件开发. E-mail: bell\_cardc@163.com. 张来平,博士,研究员,中国计算机学会(CCF)委员,主要研究领域为计算流体力学. 何 磊,硕士, 工程师,主要研究方向为 CFD 软件开发、自动化测试. 何先耀,硕士,助理研究员,中国计算机学会(CCF)会员,主要研究方向为高超声速空气动力学、计算流体力学软件开发. 郭永恒,博士,助理研究员,主要研究方向为理论与计算流体力学. 徐庆新,博士,研究员,主要研究 领域为并行 CFD 软件开发.

the requirement of the large-scale parallel flow simulations. Structured grid, unstructured grid and overset grid are all integrated into this framework. In order to ensure the grid independence and to isolate the parallel data communication operation from the physics applications, a unified communication model and a 'three-in-one' data communication pattern based on grid zone are developed. The neighboring information of multi-block structured grid zones, the interface information of parallel domain-partition zones, the 'hole-cutting' assembling relations of overlapping zones, the interface information of structured/unstructured grids are all abstracted as an data-mapping relationship model in the base level of database, and then the data communication between different types of zones can be handled in a unified way, which is independent with the grid types and the data types. In addition, according to the characteristics of parallel environment of current supercomputer systems, such as 'TianHe-2', an MPI/OpenMP hybrid parallel computing method is developed in a 'coarser-grained' way, which means that the MPI model runs between partitioned zones (regions), and the OpenMP model runs between sub-zones in each partitioned zone (region). Furthermore, both the parallel domain partition method of structured and unstructured grids, the parallel preprocess of very large-scale grids (such as the computation of cell-to-wall distance) and large scale parallel file storage model are introduced in this paper. Optimization technologies of computational efficiency, including serial array memory access and parallel zones' data packing between different computer nodes, are introduced also. Tests indicate that after optimization, both the serial array access time and the number of MPI communication are drastically reduced. Finally, three engineering application cases, including TrapWing and JSM geometries from AIAA high lift prediction workshop and wing-body configure of CHN-T1, with structured grid, unstructured grid and overset grid are tested respectively. The numerical results demonstrate the flexibility of the parallel CFD framework. On a hybrid grid with 3.32 billion cells, the parallel efficiency is higher than 95% for 0.1 million CPU processors on Tianhe-2 (based on 3072 processors) and more than 83% for 0.16 million CPU processors on CARDC in-house computer system (based on 2048 processors). Most importantly, parallel data communication used by the above three cases are exactly the same method and even the codes, which reveals that the current parallel architecture is adapt to arbitrary grid types.

**Keywords** parallel computation; MPI/OpenMP hybrid parallel computing; parallel CFD framework; PHengLEI; PHengLEI software; HyperFLOW

# 1 引 言

计算流体力学(Computational Fluid Dynamics, 简称为 CFD),是一门利用计算机和数值方法对流 体力学问题进行数值模拟和分析的新兴交叉学科. 随着计算机科学技术的不断发展,这门学科已经成 为与风洞试验和飞行试验互为补充的三大研究手段 之一,被广泛应用于飞行器设计过程之中.

CFD 属于典型的计算、访存密集型的科学计算 领域,强烈地依赖于高性能计算机的发展.早期 CFD 基本上都是采用串行方式执行,使用的计算方 法也相对简单,模拟的外形亦受网格生成技术的限制只能模拟较为简单的外形.20世纪90年代开始, CFD 在计算方法、湍流模型等方面取得很大进展, 空间和时间分辨率不断提高,网格生成技术的发展 也使得几何构型的模拟越来越真实.然而,随着计算 规模的不断扩大,计算量也呈几何量级的增长.近十 年来,得益于计算机硬件和并行计算技术的发展, CFD 软件已通过代码并行化实现了大规模数值模 拟.时至今日,以 MPI、OpenMP 等为代表的并行编 程环境已在 CFD 数值模拟中得到广泛应用.在工程 应用领域,常规气动力计算已普遍采用百核、千核进 行并行计算<sup>[1]</sup>. 目前成熟的计算流体力学方法基本上都是基于 计算网格进行的.通过计算网格,将待模拟的空间区 域离散为不同的网格单元(如四面体、六面体),然后 再基于离散后的网格进行数值模拟.根据离散单元 拓扑的不同,可将计算网格分为结构网格、非结构网 格、混合网格<sup>[2]</sup>.另外,如果网格单元之间存在互相 重叠的情况,还可以采用结构重叠网格或非结构重 叠网格.

对于采用共享内存的并行编程模式而言,不同 网格类型的并行计算没有太大的差别.但是如果采 用 MPI 并行通信,现有 CFD 软件基本上都是基于 区域分解,即将整个计算域的网格划分为不同的网 格子块,在每个网格子块上执行相应的 MPI 进程, 各网格块间进行 MPI 通信.然而,由于结构网格和 非结构网格,以及重叠网格的数据结构完全不同,在 网格分区、网格块连接关系等方面差异较大,因此往 往分别针对不同的网格类型设计不同的并行计算 框架.

国内外已经有不少知名的针对数值模拟领域的 并行计算框架,如 OpenFOAM、Moose、Overture、 JASMIN/JAUMIN/JCOGIN 等. OpenFOAM<sup>①</sup> 是 一款全球知名的面向流体力学数值仿真的开源并 行编程框架,基于非结构网格的有限体积方法求 解偏微分方程,以C++面向对象设计,具备良好的 可扩展性、稳定强大的底层类库、丰富的前后置处理 接口. Moose<sup>[3-4]</sup>,源于 LibMesh,是一个基于非结构 网格有限元方法的 C++并行开源类库,集成了 PETSc、Eigen 等底层数学开源库,主要面向固体力 学、弹性力学、热传导等问题,该平台提供了大量工 具包. Overture 是一套融合了网格生成、差分计算、 边界条件设定、数据存取和图形显示等功能的基于 结构网格的综合型数值计算软件并行编程框架. JASMIN/JAUMIN/JCOGIN<sup>[5-6]</sup>,是由中国工程物 理研究院高性能数值模拟软件中心开发的,分别面 向结构网格、非结构网格、无网格数值模拟的系列并 行编程框架,通过将并行通信分解为多个层次,实现 领域专家和并行计算的隔离,经过多年的发展,已在 国内积累了一定的用户群.

尽管上述并行编程框架已经得到多年的发展, 得到了广泛的用户认同,但由于结构网格和非结构 网格间存在先天的差异,这些并行编程框架基本都 是单独针对结构网格或者非结构网格开发的,难以 做到并行通信与网格无关,这对于现代 CFD 软件发 展来说不可谓是一大遗憾.目前在 CFD 工程应用 中,结构网格和非结构网格在不同应用领域均有各自的优势.例如,在低速、亚跨声速领域,由于外形比较复杂,往往采用非结构或混合网格离散,快速获得流场数据.在高超声速领域,由于热流和摩阻精确预测、激波和转捩等现象精细捕捉的需要,具有较高离散精度的结构网格往往占据主导地位.因此,如果对于结构、非结构网格分别采用不同的并行计算框架,那么同样的一套计算方法将难以适应于不同的网格类型.

针对上述问题,中国空气动力研究与发展中心 在已有的多款结构、非结构 In-House 代码基础上, 开发了一款同时适应于结构网格、非结构网格、重叠 网格的大规模并行计算 CFD 软件平台——"风雷" PHengLEI(源于作者团队开发的结构/非结构混 合求解器 HyperFLOW v1.0<sup>[7-10]</sup>).PHengLEI 既是 一个软件平台、又是一个并行计算框架.软件平台, 是直接面向工程型号应用生产气动数据的"数值风 洞";并行计算框架,是整个软件的编程框架,提供统 一的底层接口(网格、数据库、并行、数学库等),负责 将现有的各 In-House 代码整合,是软件平台的底层 支撑.

本文主要介绍 PHengLEI 的并行计算框架设 计,以及如何实现对任意网格类型的支持.这里 "任意"一词意指结构网格、非结构网格以及各种类 型的混合网络(二维三角形/四边形/多边形、三维四 面体/三棱柱/金字塔/六面体/多面体),甚至包括重 叠网格.关于结构/非结构求解器的耦合计算,不是 本文关注内容,后文未做重点介绍,读者可参考作者 以往的工作<sup>[8-9]</sup>.

# 2 PHengLEI 简介

PHengLEI 是一款具有完全自主知识产权的面向工程应用和学术研究的通用 CFD 软件平台.该平台借鉴了面向对象的大型软件设计理念,采用 C++语言编程.为了适应结构网格、非结构网格、混合网格、重叠网格等不同网格的计算,PHengLEI 的设计具有良好通用性、可扩展性的体系结构和数据结构<sup>[7]</sup>,实现了在同一个软件平台上,结构求解器和非结构求解器的集成,两种求解器可独立运行,且能进行结构求解器和非结构求解器的同步耦合计算<sup>[8•9]</sup>,即在流场中同时含有结构网格和非结构网格的情况

① OpenFOAM website. http://www.openfoam.com/

下,在结构网格上调用结构求解器,在非结构网格上 调用非结构求解器.该软件已于 2016 年面向全国行 业内单位免费发布使用(申请使用见气动中心官网<sup>①</sup>).

平台求解的是积分形式的雷诺平均 N-S方程:

$$\frac{\partial}{\partial t} \int_{a_i} \mathbf{Q} \, \mathrm{d} \mathbf{V} = -\int_{\partial a_i} (\mathbf{F}) d\mathbf{S} + \int_{\partial a_i} (\mathbf{F}_v) \, \mathrm{d} \mathbf{S} \qquad (1)$$

这里,Q是待求解的流场变量,F和F。分别表示无 黏通量和黏性通量.

$$Q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho e \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \rho V \\ \rho u V + n_x p \\ \rho v V + n_y p \\ \rho v V + n_z p \\ \rho W V + n_z p \\ \rho H V \end{bmatrix},$$
$$\mathbf{F}_v = \begin{bmatrix} 0 \\ n_x \tau_{xx} + n_y \tau_{xy} + n_z \tau_{xz} \\ n_x \tau_{yx} + n_y \tau_{yy} + n_z \tau_{yz} \\ n_x \tau_{zx} + n_y \tau_{zy} + n_z \tau_{zz} \\ n_x \Theta_x + n_y \Theta_y + n_z \Theta_z \end{bmatrix},$$
$$\Theta_x = u \tau_{xx} + v \tau_{xy} + w \tau_{xz} + k \frac{\partial T}{\partial x},$$
$$\Theta_y = u \tau_{yx} + v \tau_{yy} + w \tau_{yz} + k \frac{\partial T}{\partial y},$$
$$\Theta_z = u \tau_{zx} + v \tau_{zy} + w \tau_{zz} + k \frac{\partial T}{\partial z}$$
(2)

(u,v,w)是速度分量, $(n_x, n_y, n_z)$ 是单元面法 向分量, $V = un_x + vn_y + wn_z$ , $\tau_{ij}$ 是切向应力,T是温 度,k是热传导系数.

数值求解方程(1)时,将空间项和时间项分开 离散求解,其中无黏项采用 Roe、Vanleer、AUSM、 Steger-Warming 等格式计算;采用最小二乘法、高 斯格林法等进行梯度计算,并通过线性重构以达到 二阶精度;黏性项采用中心型格式;时间推进采用 LU-SGS 隐式方法求解;非定常计算时,采用双时 间步方法;湍流模拟采用 SA-方程、κω-SST 两方程 等湍流模型,还可进行基于上述模型的 DES 模拟. 软件的具体算法见参考文献[8].

## 3 网格分区策略

PHengLEI并行计算框架采用基于区域分解的 MPI/OpenMP 混合并行计算方案,二者均采用基于 网格块的粗粒度并行模式.为了适应不同类型的计 算网格,我们分别针对结构、非结构网格发展了网格 分区技术. 通过网格分区,将原始的整体网格划分为若干 块网格,从而使得每个进程上有一个或者多个网格 块.另外,当初始网格规模较大时,串行分区难以执 行,需要进行并行分区.以下分别介绍结构/非结构 两种网格类型的串行分区技术和并行分区技术.

#### 3.1 结构网格分区

PHengLEI采用贪婪算法(Greedy Algorithm) 对多块结构网格进行分区,基本思想是:通过多次划 分,将最大网格块的全部或部分分配到当前最小负 载进程中,使各进程单元总数基本一致,从而实现负 载平衡.主要步骤包括:

第一步. 根据网格单元与进程的总数,计算各 进程对应的"理想"平均单元个数为 Naver.

第二步. 当未分配网格集合非空时,循环执行 如下操作:

① 取当前状态下负载最小的进程 P<sub>i</sub> 及未分配的最大网格块 Z<sub>largest</sub>,其单元总数为 N<sub>cell</sub>.

② 设  $Z_{\text{largest}}$ 未经切割直接分配到第  $P_i$ 进程,则 该进程的预期负载为  $\tau = N_{\text{load}} + N_{\text{cell}}$ .其中, $N_{\text{load}}$ 为 第  $P_i$ 进程的已有单元总数.可能超出平均负载的单 元个数为  $N_{\text{rest}} = \tau - N_{\text{aver}}$ .第  $P_i$ 进程计划分配的单 元个数为  $N_{\text{target}} = N_{\text{cell}} - N_{\text{rest}}$ .

为了确保划分后的网格块不至于过小,需要避 免碎片的出现.这里令网格碎片块的维度为 $N_{min} = 3 \times 3 \times 2$ ,负载平衡相对误差为 $\epsilon = 0.05$ ,若 $\tau > N_{aver}(1.0+\epsilon)$ 、 $N_{rest} > N_{min}$ 、 $N_{target} > N_{min}$ 同时成立,执行③,否则执行④,

③对块  $Z_{largest}$  进行切割,计算剖分相对百分比 参数  $\gamma = N_{target}/N_{cell}$ ,若 0.25  $< Z_{largest} <$  0.75,则在  $Z_{largest}$ 块上沿着单元维数最大的方向(不含极轴)进 行切割,使新生成的左块分配到第  $P_i$ 进程中,右块 留在未分配网格块集合中;否则,沿着同样的方向, 在中位面处切割,使左右块所含单元个数基本相同, 但二者皆留在未分配网格块集合中.特别地,对多重 网格,实际切割面应选取距上述切割面最近的位置.

④ 网格块 Zlargest 直接分配至第 Pi 进程中.

当上述循环过程结束时,未分配网格集合变为 空集,网格分区完成.图1中以一个二维网格为例, 说明贪婪算法分区过程.图2是客机外形的结构网 格物面分区图.

① China Aerodynamics Research and Development center, PHengLEI software. http://www.cardc.cn/phenglei/ phenglei.html

2372

计 算 机 学 报







图 2 飞行器结构物面网格分区

## 3.2 非结构网格分区

非结构网格分区的内核是开源分区软件 METIS<sup>[11]</sup>,基本思想是采用基于图的多级分区算 法<sup>[12]</sup>.基于图(Graph)的多级分区算法分三步完成 (如图 3):①图的粗化(Coarsening, $G_0 \sim G_4$ ),网格 单元聚团,把区域分为大粒度的网格区域;②初始 化分区(Initial, $G_4$ ),对聚团后的大粒度网格区域进 行初始分区;③多级优化分区( $G_4 \sim G_0$ ),对初始分 区的结果逐级进行优化分区.



图 3 非结构网格分区:多级分区算法[11]

从多级分区算法的分区过程可以看出,它可以 很好的保证子区域的单连通性,同时由于在第三步 中是基于较粗的粒度从全局来考虑,可以运用一些 较好的算法,不仅能够做到任意分区,而且可以较好的做到各子区域的负载平衡,并且能够有效地减少 子区域之间的边界单元,这些性能在并行计算中都 可通过大大减少边界通讯量而提高并行计算效率.

采用 METIS 分区后,获得了原始全局网格中 每个单元在拟分区局部网格中的编号,根据该编号, 从原始网格中分别重构出分区后的网格点、面、单 元,完成整个非结构网格的分区.

## 3.3 并行分区技术

上文中分别介绍了针对结构、非结构网格的分 区技术,算法均是串行执行.当网格量不是太大时, 串行的分区方式能满足需求,但是当网格量大幅增 加到十亿乃至上百亿后,串行分区的效率将急剧下 降,甚至可能因为内存限制而分区失败.针对该问 题,PHengLEI发展了并行网格分区技术.

基本思想是,在原始网格上采用上述串行算法 划分出少量的子分区,然后在这些子分区上并行分 区(即每个进程分别在每个子分区执行串行分区算 法),并利用 MPI 通信并行匹配每个分区的交界面 信息.图4用一个简单的二维非结构网格为例说明 并行分区过程.需要说明的是,图中的 Step3 可以重 复执行以快速完成超大规模网格的并行分区.例如 要将初始网格分为 1024 个分区,可以按照"1 分 2、 2 分 8、8 分 64、64 分 1024"的顺序(1 分 2 是串行、 其它为并行执行),快速实现百万量级网格块的分区.

与串行分区算法不同的是,在进行最终的子区 域网格的重构过程中,由于每个进程中只有当前 子块的交界面信息,而缺失相邻进程中网格块的交 界面,需要通过 MPI 通信获得.为此,采用了一种 "局部全局化"的策略.如图 5,如果要重构目标块*i* 的交界面,则将其邻居(*i*<sub>0</sub>~*i*<sub>3</sub>)中的所有交界面信息 发送到目标块*i*,使得对于目标块*i*来说,好像拥有 了"全局"的交界面一样,进而实现交界面的快速重 构.在面重构中用到了哈希数据结构,以搜索具有相 同几何点的面.





图 4 并行分区过程

图 6 是客机外形并行分区后物面上分区边界对 比图,左侧为 64 分区、右侧为 1024 分区,可看到,右 侧的所有子分区是左侧分区的子集.与串行分区相 比,并行分区不仅能解决内存制约的问题,还能大幅 提高分区效率.表1是串行与并行分区的效率对比, 如果用串行分区,随着分区数的增加,计算时间呈指 数增加,而并行分区则可以通过增加进程的方式来 有效解决这一问题.



图 6 客机物面并行分区

主 1 安扣施责并纪八页			
	<b>±</b> 1	它扣临声并仁八回	

	1024 分区时间/s	8192 分区时间/s
串行分区(1进程)	1442	10789
并行分区(64 进程)	7.13	
并行分区(1024 进程)	_	12.48

# 4 通用并行框架设计

PHengLEI设计了能兼容包括结构、非结构和 重叠网格的任意网格类型的并行计算框架,下文分 别从通信模式、分层结构、混合并行等几方面介绍并 行计算框架的设计,并说明如何实现与网格类型无 关的并行化编程.

## 4.1 通信模式

PHengLEI采用了基于循环遍历所有网格块的 模式进行通信:每个进程均遍历所有网格块(仅仅是 块编号),当所遍历的网格块属于当前进程时,则 向邻居网格块所在进程发送数据;当所遍历的网格 块属于其邻居网格块时,则从邻居网格块所在进程 接收数据.图7是通信模式示意,图中给出了4个进 程、4个网格块,将每个网格块分配至具有相同编号 的进程(当然实际问题可能是1个进程中有多个网 格块).以1号进程为例,遍历4个网格块,该进程上 只有1号网格块(与3号进程中的3号两格块为邻 居关系),当遍历到1号网格块时向3号进程发送消 息,当遍历到3号网格块时从1号网格块接收消息, 而对于0号、2号网格块则跳过.这里,每个进程可



图 5 并行重构交界面过程:"局部全局化"

见的全局网格块编号实际上是局部的,即只有网格 块或者其邻居位于本进程时才遍历,以适应大规模 并行计算要求.

上述通信模式中的网格块,既可以是结构网格 也可以是非结构网格,对于重叠网格也同样适用,每 个进程中可以是一块也可以是多块.通过基于网格 块的通信循环遍历通信,将通信与网格类型隔离,实 现网格类型无关性.

#### 4.2 并行计算框架分层结构

图 8 是并行计算框架的分层结构,由上至下分 别是:并行 API 接口层、交界面数据层、数据底层 (SuperContainer)、MPI 通信层.



图 8 进程、网格块、交界面间的关系

### 4.2.1 并行 API 接口层

并行计算框架的最顶层是 API 接口层,二次开 发者通过调用.h 头文件和静动态库,只需要编写串 行程序,就能自动完成并行求解器开发,设计目的是 使用户无须深厚的并行编程基础,就能完成并行计 算求解器开发.

求解器	通过 API 接口注册要通信的数据	
Solver	求解器基类	
CFDSolver	定义注册机制	
NSSolver	CFDSolver::RegisterInterField("q", double, 5) CFDSolver::RegisterInterField("t", double, 1)	
NSSolver_Unstr	CFDSolver::RegisterInterField("dqdx", double,5) CFDSolver::RegisterInterField("dqdy", double,5) CFDSolver::RegisterInterField("dqdz", double,5) CFDSolver::RegisterInterField("limiter", double, 1)	

+	•	ᅶᄻᆔᄜ	1.01	1÷ —	14 /7.1
表	2	<b> </b>	API	接口	14月11月11月11月11月11月11月11月11月11月11日11月11日11日

表 2 中以 CFD 求解器为例,说明 API 接口调用.在 PHengLEI 框架中,Solver 类是所有求解器的基类,派生出 CFD 求解器 CFDSolver 类,继而分别派生出 N-S 方程求解器 NSSolver\_Str 类和非结构 NS 求解器 NSSolver\_Unstr 类.在 CFDSolver 类中定义了并行数据注册机制.不管是结构还是非结构的 NS 方程 求解器都需要通信流场变量 q 和温度 t 等物理量, 而非结构求解器还需要通信梯度、限制器系数等数据.各求解器调用 CFDSolver 中定义的 Register-InterField 接口,给定数据名、类型、维度,就可完成 该数据的并行注册操作;后期计算中,并行框架的底 层负责完成已注册数据的并行通信.

4.2.2 交界面数据层

交界面数据层,即几何网格分区后形成的交界 面信息,其中存储了需要通信的数据.

通信模式是整个计算域中所有计算资源间的信息交换模式,具体到网格块来说,则是网格块间的交 界面数据交换.

PHengLEI 框架基于结构/非结构网格的有限 体积方法求解 N-S方程,基于区域分解的方法进行 并行计算. 区域分解算法,即采用上文所述的结构/ 非结构网格分区方法将原始网格划分为多个子网格 块,形成的子网格块分配至多个进程,各子网格块间 形成几何交界面(图 9),通过交换交界面上的变量, 完成多块网格并行计算.特别地,若一个进程内有多 个网格块(结构多块网格)或在节点内进行 OpenMP 并行,则网格块间的数据直接交换;若相邻网格块处 于不同进程,则采用 MPI 进行信息通信. 对于重叠 网格,实际是通过重叠区"洞"边界插值进行数据交 换,其原理与"点对点"数据通信一致,可以采用同样 方法处理.



图 9 子网格块间形成的交界面及通信

图 10 是进程、网格块、交界面间的关系,图中 有两个进程 i 和进程 j,每个进程上假设分别有两个 网格块(实际可能有任意多块). zone  $i_0$ /zone  $j_0$ 、 zone  $i_1$ /zone  $j_1$ 分别构成邻居关系,二者间的交界面 信息需要通信交换数据,即 zone  $i_0$ 的交界面信息 Interface  $i_0$ 要与 zone  $j_0$ 的交界面信息 Interface  $j_0$ 之间进行数据通信, zone  $i_1$ 的交界面信息 Interface



图 10 进程、网格块、交界面间的关系

 $i_1$ 要与 zone  $j_1$ 的交界面信息 Interface  $j_1$ 之间进行 数据通信.

结构网格和非结构网格最大的区别是网格几何 信息数据结构.对于非结构网格,计算域中的点、面、 单元均为无序存储,采用基于"面"的数据结构存储 拓扑连接关系,主要有:"面-点"关系、"面-左右单元" 关系;对于结构网格,计算域中的点、单元均按;为本方 向有序存储,其拓扑连接关系默认隐含在*i-j*本方向 中.基于封装性和效率的综合考虑,PHengLEI中, 采用一般的C++指针数组存储非结构网格数据, 同时借鉴了Blitz 库封装了一种类似于 FORTRAN 语言的数组类型来存储结构网格数据.本文主要阐述 并行计算方法,相关的数据结构设计可参考文献[7]. 4.2.3 数据底层

数据底层,存储并交换网格块间的几何数据、流 场数据.通过数据底层,可以在完全相同的机制下实 现以下数据信息交换:

①进程内的结构多块网格交界面数据;

②节点内任意网格块交界面数据,主要用于 OpenMP并行化和结构/非结构耦合计算;

③进程间的相邻网格块交界面数据,主要用于 MPI通信.

④结构网格与非结构网格块间的数据交换.

为了实现以上信息交换功能,PHengLEI设计 了一种基于数据打包传输的机制,以及一种能适应 于任意类型数据的"超级容器":SuperContainer. 图 11 是网格块/交界面/数据交换机制图,其中的网 格块(zone)可以是任意类型的网格,整个数据通信 包括三个过程:数据压缩一MPI通信一数据解压.

数据压缩:如图 11 上,进程 *i* 和进程 *j* 分别将 其上的所有网格块交界面信息压缩到"超级容器" data,目的是将两两进程间需要通信的数据打包,即



图 11 底层数据通信:压缩(左)、解压(右)

能通过封装实现任意类型数据的交换又能减少通信 次数.图 12 是对于 32768 块网格、2048 进程情况 下,是否打包通信的通信次数对比,可以看到,数据 打包通信后,总的通信次数有量级上的减少.



MPI通信: 将打包到"超级容器"中的数据,采用 MPI通信.由于 CFD 数值模拟基本上都是采用迭代 方法进行,因此在每一个迭代步均需要进行通信.图 13 是迭代计算与 MPI 通信时序,每进行完一步迭代 计算后,在求解器的后处理过程中进行 MPI 通信.



图 13 迭代计算与 MPI 通信时序

数据解压:如图 11下,将 MPI 通信后得到的相邻区的网格块交界面数据解压,从而获得当前网格块的新邻居边界信息.

在这种数据交换模式中,将任意类型网格块的 交界面数据按照接口标准压缩到"超级容器"数据 中,不同进程间只进行"超级容器"数据的并行通信, 将数据结构隔离在并行通信的底层,从而实现任意 类型网格的并行计算兼容性.

4.2.4 MPI 通信层

MPI通信层,是具体的执行消息传递的接口层,内部调用 MPICH/MSMPI等不同的 MPI 库,通过将通信接口与具体的实现方式隔离,提高跨平台移植能力.

## 4.3 MPI/OpenMP 混合并行模式

在阐述 PHengLEI 的混合并行模式前,有必要

先了解 CFD 软件的主要运行特征.图 14 是非结构求 解器计算全过程的运行性能,图中只监控了若干步 迭代步过程.整个运行分三个阶段:Stage1,若干个服 务器进程读取分组的网格文件,并发送至相对应的 计算进程;Stage2,计算壁面距离;Stage3,CFD 迭代 计算.可以看到,PHengLEI 在计算阶段(Stage3), CPU 计算和内存访问都处于高负荷状态,这也是 CFD 软件的普遍特点:计算密集、频繁访存,尤其是 对于非结构求解器,由于非结构网格数据的随机无 序存储,导致访存问题更突出.



图 14 PHengLEI 运行性能监测

近年来,超大规模高性能计算机集群飞速发展, 在 Top500 排行榜前列的 HPC 几乎都是异构集群. 异构计算机对 CFD 软件带来了挑战,主要表现在代 码封装性、内存访问等方面.另外,目前的多种异构 架构互相差别较大,也为 CFD 软件开发者在硬件适 配方面带来了很大的不确定性.虽然一些研究型代 码在"天河二号"、"神威•太湖之光"等异构 HPC 上 进行了测试,但是对于工业应用部门的生产型 CFD 软件(如 PHengLEI),为了保持其代码的可扩展性、 易读性、可维护性,基本都没有进行异构并行改造, 而且其硬件环境还基本上都是定制的多核处理器.

当前,CFD软件通常运行在分布式集群上,采 用 MPI 进行进程间的通信.对于多核处理器来说, 节点内部的内存是共享的,如果在节点内部也采用 MPI 通信,可能会由于带宽的限制带来通信总量 的增加,采用单一的 MPI 通信往往无法得到较优的 效果.其中一种解决方案是采用 MPI/OpenMP 混 合并行,即在节点内部用 OpenMP 并行,而在节点 之间用 MPI 并行,从而实现节点间和节点内部的两 级并行,充分利用消息传递和共享内存两种编程模 型的优点.

一般来说, MPI/OpenMP 混合并行有两种方案<sup>[13]</sup>:(1) 细粒度混合并行,即对网格块内部的热点

循环代码进行 OpenMP 并行,而在网格块边界上用 MPI 并行;(2) 粗粒度混合并行,即采用两级分区并 行的方案,在一级分区上用 MPI 并行,然后将一级 分区划分为若干个二级分区,二级分区的数量一般 小于等于节点内部的核数,在二级分区上进行粗粒 度的 OpenMP 并行.

PHengLEI,既是一个二次开发并行开发框架, 又是作为气动数据生产的"数值风洞",要求代码将 并行隔离至底层,同时保持代码的可扩展性、稳定 性.如果采用细粒度的循环级 OpenMP 并行,将导 致软件的可扩展性、可维护性受到破坏,影响多人协 同开发.因此,在综合衡量效率和封装性等因素后, 采用了基于网格分区的粗粒度两级并行.

在 PHengLEI 中,结构网格和非结构网格的混 合并行都采用相同的基于网格二级分区的粗粒度模 式. 但是与非结构网格不一样的是,结构网格的第二 级分区要做到负载平衡有一定难度,目前正在开展 结构网格二级分区负载平衡相关的工作.

表 3 是对于 5100 万非结构网格,一级分区(对应 MPI 进程)分别为 64 到 1024,二级分区(对应 OpenMP 线程)分别为 1 到 16 情况下,MPI/OpenMP 混合并行计算 100 步的时间对比.可以看到,当二级 分区(线程数)为 4 时,混合并行相比于纯 MPI 并行

有一定优势.但在其他情况下,混合并行的优势并没 有得到体现.

_				
	节点数	进程数	每节点内使用线程数	计算时间/s
	64	64	16	159.91
	64	128	8	144.91
	64	256	4	141.04
	64	512	2	145.88
	64	1024	1	142.99

表 3 混合并行时计算时间比较

小规模网格下,通信时间绝对值较小,为了进 一步分析通信时间,对更大规模的 CFD 计算进行 了测试.表4 是采用 33.2 亿超大规模网格,分为 32768 个网格分区进行计算的通信时间测试,其中 给出了 100 步迭代过程中的每一步的最大、最小通 信时间.可以看到,对于阻塞通信而言,当线程数增 加、进程数减小后,通信时间减小,混合并行比纯 MPI 并行加速约 13%;对于非阻塞通信,混合并行 并没有明显优势.

表 4 通信时间比较

进程×线程数	阻塞通信/s	非阻塞通信。
$16 \times 2048$	1.36~1.46	0.17~0.35
$8 \times 4096$	1.35~1.45	0.20~0.32
$4 \times 8192$	1.49~1.59	0.08~0.20
$2 \times 16384$	1.49~1.56	0.10~0.30
$1 \times 32768$	1.58~1.69	0.08~0.15

图 15 是每个 MPI 进程的最大/最小/平均通信 次数和整个作业的通信次数统计,这里的通信次数 是指每个 MPI 进程与其他进程的信息交换次数.当 线程数增加、进程数减小后,由于进行了打包通信, 使得通信量减少,但相应的计算时间并没有减少,这 是由于系统通信带宽很大所致.实际上,上述测试应 用的 HPC 系统节点间的通讯带宽为 10 G 级,对于 CFD 并行通信过程而言, MPI 通信数据总量最大仅



图 15 MPI 通信次数统计

为 MB级,相对于带宽来说是小量,这也可能是混合并行的优势并不突出的原因.在带宽较小的集群上, 混合并行可能体现出更大优势.我们将在后续工作 中对这一问题进行进一步的分析.

# 5 面向 CFD 应用的并行文件存储模式

随着计算机运算速度的不断提高,CFD 对几何 外形细节模拟越来越逼真,网格量大幅增加,目前在 实际飞行器构型气动数据生产中的网格规模几乎都 是千万甚至上亿量级.另外,随着计算方法的不断发 展,诸如 DES、LES/RANS等湍流模型和高精度方 法的应用,使得空间流场数据大幅增加,这些因素导 致流场文件越来越大,每个计算状态的流场数据达 到 GB 甚至 TB 级.对大规模气动仿真而言,计算状 态往往达到成千上万个,流场文件的大小和数量急 剧增加,设计合理的文件存储及 IO 模式,能在很大 程度上保证计算的效率.

一般来说,大规模 CFD 并行计算时的文件存储 有两种模式.一种是服务器模式,即用一个服务器进 程负责所有进程的数据及文件 IO,所有的数据存储 为一个文件;另一种是对等模式,即每个进程负责管 理各自的数据及文件 IO,文件数等于进程数.

在 CFD 应用中,这两种模式各有优缺点.对于 服务器模式,文件只有一个,CFD 可视化软件能快 速加载流场数据,工程中的文件数量少,不会影响文 件系统的平衡,但是当网格规模较大时,单个文件容 量很大,可能会达到 TB 量级,导致 IO 时间过长; 对于对等模式,每个文件大小有限,但是会造成文件 数量过多,一个工程中就有可能达到十万、百万量级 的文件数,可能会破坏文件系统的平衡,导致文件系 统"卡死".

针对上述问题, PHengLEI 计算框架中设计了 一种基于分组的文件存储模式.图 16 是并行文件分 组存储示意图, 在多个进程并行计算的情况下, 将所



图 16 并行文件分组存储

有进程分为若干个组,每个组中指定一个进程为该组的服务器进程,该服务器进程负责管理该组中的所有进程的文件.并行计算过程中,每个进程将需要 IO 的文件数据先压缩到 3.2.3 中设计的"超级容器"中, "超级容器"负责将数据发送至所属的该组服务器进程,由该服务器进程集中 IO.

实际上,这种分组的并行文件存储模式是上述 两种文件存储模式的通用情况:当分组数为1时,即 是服务器模式;当分组数等于进程数时,即是对等模 式.分组存储的优势很明显:在网格量、并行规模都 很大的情况下,分组存储后,只需通过若干个文件就 能存储流场数据,既不会使文件数过多,又不会使单 个文件过大,有利于 CFD 可视化、数据提取等后处 理. 更为重要的是,在 CFD 迭代计算过程中,每隔一 定的步数就需存储流场文件,采用这种并行存储模 式可以加速 IO 过程,提高整体并行效率.表5 是单 个文件串行与分组文件并行 IO 的时间对比,分组 并行读取网格文件的时间比串行读取有量级的减 少,表中仅是一次文件 IO 时间,在 CFD 迭代计算过 程中会频繁读写,整体时间将大幅减少.当然,对于 不同大小的文件和不同的 IO 环境,分组策略应该 存在一个最优参数,我们将在他文中详细讨论,这里 不再赘述.

	1024 分区时间/s	8192 分区时间/s
串行	6.83	11.93
并行(64 进程)	1.15	_
并行(1024 进程)		0.92
测试环境:FT2000 处	理器,每节点16核,M	MPICH 3.1.3版本.

表 5 网格文件并行 IO 时间比较

# 6 串行和并行效率优化

## 6.1 串行优化

串行计算效率是整个计算的基础,PHengLEI 平台采用 C++编程,由于 C++中没有内置多维数 组结构,为适应结构网格求解器对多维数组的需要, PHengLEI 封装了类似于 FORTRAN 数组的数据 结构 FYArray,用于存储多维数据.FYArray 的底 层依然是一维连续存储,通过记录下每一维的跨度, 从而完成数据的索引.通过首地址的偏移,使得数组 下标的起始不为 0.以三维数组为例(图 17),假设每 维的跨度分别为  $s_0$ , $s_1$ , $s_2$ ,那么存取第(i,j,k)个元 素就是首地址后第( $k \times s_1 \times s_2 + j \times s_2 + i$ )个元素. 然而,PHengLEI 在求解过程中有大量循环计算,在 循环最里层都会进行数据的存取,每一次的存取都 要进行偏移地址的计算,从而耗费大量时间.在保证 程序的封装性的前提下,为了提高访存效率,采用了 两种优化策略:数据多维化和数据一维化.





### 串行优化策略 1:数据多维化

采取了以空间换时间的优化策略,类似原始的 C++多维数组,层层记录下每个维度的索引地址 (此例图中的 addr0,addr1),通过两次寻址替代偏移 地址的计算完成元素的存取.相应的,整个数组结构 增加了  $s_0 \times s_1$ 个地址存储空间.

### 串行优化策略 2:数据一维化

进一步通过循环计算部分的汇编代码分析发现,大部分循环都是相邻元素的连续数据存取,在每 一层循环中存在地址的重复计算问题.因此,针对循 环计算进行了进一步的优化,将元素的索引计算分 割到每一层中,通过计算偏移地址长度,最内层元素 直接访存,避免了每一层循环中相同地址跨度计算 开销.

使用维度大小为100×500×1000的数组,进行 访存测试,对多维数组原始访存、多维化访存、一维 化访存三种策略进行对比.从图18的测试结果看, 三者耗时比例大概为8:3:1,两种优化效果非常明 显.三种访存方式对代码的封装程度与优化效果成 反比,即效率越低、封装程度越好.在实际编码时,可 以首先用原始访存方式,以类似FORTRAN数组结 构的方式方便编程,代码成熟后,将数组访存改为一 维或多维化访存,以提高计算效率.



## 6.2 MPI/OpenMP并行优化

PHengLEI采用 MPI/OpenMP 混合并行通信 方案,通信过程中,进程内部各网格块的数据先打 包,然后在进程间进行打包后数据的 MPI 通信. 图 19 是数据打包通信过程各步骤耗费的时间,可以 看到,数据压缩/解压的时间比核心的 MPI 通信还 多,存在热点.



经分析后,采用两个优化手段:(1)节点内部数 据不打包.由于节点内部的数据不需要 MPI 通信, 因此不需要进行打包,各线程直接读取所需的数据 即可;(2)压缩/解压并行化.优化前数据压缩/解压 是串行执行的,即将各网格块数据逐个、依次压缩 (或解压),此时节点内的其他线程处于空闲状态,采 用 OpenMP 使数据压缩/解压并行化.图 20 是采用 OpenMP 对数据压缩/解压并行化后,数据通信一次 的性能对比,可见,优化后数据处理时间大约减少为 原来的三分之一.



7 CFD 应用测试

下面分别基于非结构网格、对接结构网格、重 叠结构网格,对典型工程应用级 CFD 算例,测试 PHengLEI并行计算框架的并行模式.本文只介绍 算例的并行性能,算例结果的验证与确认内容部分 已在文献[9-10,14]中介绍.

除了 7.1 节以外,其它的应用测试均使用国产 集群,全部为飞腾 2000 多核微处理器芯片,每个节 点 16 核,共享内存 32G,10G 级通信带宽.

# 7.1 非结构网格算例:客机起降状态

对于大型客机外形来说,在起降状态飞行攻角 较大,由于背风区大分离涡的存在,采用较稀疏的计 算网格难以捕获空间细致结构,需要采用大规模网 格进行数值模拟.

目前常用的商用网格生成软件仅支持网格串行 生成,对非结构/混合网格而言,一般最大规模限定 在1亿量级左右,难以满足超大规模非结构网格生 成的需求.为此,采用作者开发的大规模并行网格生 成算法<sup>[15]</sup>,基于 N 个进程并行生成数十亿规模的 网格,因此最小计算进程数为 N.并行网格生成的 基本思路是首先利用商用网格生成软件生成数千万 的初始网格,然后将初始网格分区,在分区并行基础 上利用全局并行自适应生成超大规模网格;为了确 保几何保形,对自适应后的物面网格利用原始 CAD 数据投影;最后对投影后的边界层内网格,利用径向 基函数(RBF)并行动网格生成方法进行优化,确保 边界层内网格的质量.

图 21 是客机高升力构型及非结构网格局部. 计算中采用了粗、中、密三套网格,单元量分别为 0.5 亿、4.2 亿、33.2 亿.由于计算资源有限,密网格 流场模拟未计算收敛,仅进行了并行效率测试.分别 在国产定制集群和"天河二号"上进行了大规模并行 测试,表 6 是两个 HPC 的运行环境对比.



图 21 客机标模计算外形及非结构网格局部

表 6 集群比较

	国产定制集群	天河二号
CPU	FT2000	Intel Xeon E5-2692
	16 cores per node	24 cores per node
内存	32G per node	64G per node
操作系统	Kylin	Kylin
编译器	Phytium c++ 1.0.0 Based on GNU GCC 4.9.3	Intel compilers-11.1
MPI	MPICH 3.1.3	MPICH 3.1.3

图 22 是在两个集群上的测试结果,在"天河二号" 上进行了近 10 万核测试,在国产定制集群上进行了 16.38 万核测试.可以看到,在 33.2 亿网格规模时, 国产集群在 163840 核下并行效率保持在 83%以上 (基于 2048 核),"天河二号"上在近 10 万核下并行 效率保持在 95%以上(基于 3072 核).



2380

图 22 两个集群并行效率:定制集群(上、天河二号(下)

该算例用到了前文所述的并行分区、分组存储 等技术,这里以在"天河二号"上的计算为例说明. (1)并行分区:并行生成的初始网格分区数为192, 然后将这些分区进一步划分为3072~98304个子 分区;(2)分组存储:对于33.2亿网格而言,仅网格 文件大小就有204G,加上流场文件会成倍增加,如 果不用分组存储,会使得文件过大.为此,计算中将 所有存储文件分为192组,每组文件管理512个网 格块及对应的流程文件,每个文件大小约为1G.通 过并行分区、分组存储,有效地解决了超大规模网格 及流场数据的存储问题.

图 23 是该外形的升力系数. 从图中可以看到, 采用加密的网格、大规模的并行计算后,结果与试验 值符合更好<sup>①</sup>,充分证明了大规模高性能计算对实



际工程应用的推动作用,表明了 CFD 应用领域对高 性能计算机的实际需求.图 24 是 4.2 亿网格的大攻 角状态下 DES 模拟的流场结构(Q 准则等值面).如 此规模的实际外形 DES 模拟在文献中尚不多见.该 算例说明了 PHengLEI 并行计算框架对非结构网 格的适应性.



图 24 流场结果:涡量等值面

### 7.2 结构网格算例:客机标模巡航状态

对于新一代大型客机来说,气动力系数尤其是 阻力系数的精确预测,对燃油经济型、航程等有至关 重要的作用,因此需要 CFD 计算时尽可能提高空间 分辨率,即需要使用大规模网格量.这里采用结构网 格离散客机标模.

图 25 是客机的外形、垂尾局部结构网格和空间 的网格几何拓扑,整个网格量约 31 亿单元,对巡航 状态进行模拟,飞行马赫数 0.73.图 26 是并行计算



图 25 客机标模外形及局部结构网格



① AIAA high lift work shop. http://hiliftpw.larc.nasa.gov

效率,采用贪婪分区算法,将原始网格分别分为 4096 至 32768 区,可以看到,PHengLEI 在 32768 核 下基于 4096 核的并行效率能保持在 60%以上.需 要说明的是,对于结构求解器,由于结构网格分区无 法做到完全的负载平衡,而且进程内多个网格块间 频繁交换数据,因此其并行效率较非结构网格低.该 算例说明了 PHengLEI 平台适用于工程外形的结 构求解器并行计算.

### 7.3 重叠网格算例:高升力外形

高升力外形是客机、运输机等大型飞行器的简化研究外形,为了评估 CFD 技术对高升力外形的预测能力,为工程应用提供经验指导,美国 AIAA 选用了NASA 梯形翼高升力构型作为主要研究对象<sup>[10]</sup>.这里采用重叠网格模拟高升力外形,用 PHengLEI 重叠结构网格并行求解器进行模拟.

图 27 是高升力外形重叠网格拓扑结构,共 1159 万单元,309 个结构网格块.图 28 是并行效率 图,512 核计算时并行效率在 40%左右.由于重叠网 格算法较为特殊,并行计算过程中,除了要进行边界 条件处理外,还需要进行整个计算域的重叠网格插 值,尤其是在考虑运动重叠网格的情况下,还要进行 插值关系搜索,因此一般来说重叠网格求解器的并 行效率都不高.后续我们将针对这一问题,开展进一 步的并行优化,并开展大规模的并行计算测试. 图 29 是局部重叠网格视图和空间的压强云图.



图 27 高升力外形重叠网格拓扑





图 29 局部重叠网格及空间压强云图

# 8 结束语

在结构/非结构网格区域分解基础上,通过设 计包含并行 API 接口层、交界面数据层、数据底层 (Super Container)、MPI 通信层的分层并行计算框 架,实现了同一个框架上对任意网格类型 CFD 计算 的兼容,同时将并行数据交换隔离到底层,做到了领 域专家编程与并行化的无关性.此外,还发展了针对 大规模 CFD 计算的并行分区、并行文件 IO 及存储、 MPI/OpenMP 混合并行等技术,提高了软件的可 扩展能力.分别采用基于非结构网格、对接结构网 格和重叠结构网格的算例,进行了实际工程外形的 并行计算,体现了较好的并行计算效率,展现了 PHengLEI 并行计算框架对任意网格类型的适 应性.

一直以来,高性能计算机硬件的发展引领着 CFD 行业应用的发展,尤其是近年来异构体系计算 机飞速发展,已经成为未来的发展主流.然而,CFD 软件的发展往往滞后于硬件的发展,尤其是在应用 行业数据生产部门,软件适应硬件的前提是要能满 足现阶段的应用需求.目前来看,全球有多种异构 HPC 架构在迅速发展,既为 CFD 软件的发展带来 了机遇,也带来了诸多挑战.当前气动中心的 HPC 系统仍然采用的是纯 CPU 分布并行架构,这也正 是 PHengLEI 软件平台到目前为止还没有进行异 构并行改造的原因之一.着眼未来计算机体系结构 的变化,下一步将重点开展异构并行计算技术的研 究,并对并行性能持续进行优化,尤其是实际构型的 结构网格并行计算效率优化.

**致 谢** 原气动中心赫新研究员在风雷软件开发过 程中做了开创性工作,何琨、陈亮中亦有贡献,在此 一并感谢!



- [1] Slotnick J, Khodadoust A, Alonso J, et al. CFD vision 2030 study: A path to revolutionary computational aerosciences. NASA/CR-2014-218178, 2014
- [2] Baker T J. Mesh generation: Art or science? Progress in Aerospace Science, 2015, 41: 29-63
- [3] Tonks M, Gaston D. An object-oriented finite element framework for multiphysics phase field simulations. Computational Materials Science, 2012, 51(1): 20-29
- [4] Williamson R L, Hales J D. Multidimensional multiphysics simulation of nuclear fuel behavior. Journal of Materials, 2011, 423(1-3): 149-163
- [5] Guo Hong, Mo Ze-Yao, Zhang Ai-Qing. A parallel module for the multiblock structured mesh in JASMIN and in applications. Computer Engineering & Science, 2012, 34(8): 69-74(in Chinese)

(郭红,莫则尧,张爱清. JASMIN 框架中多块结构网格拼接 并行计算及其应用. 计算机工程与科学,2012,34(8):69-74)

- [6] Sun Yun-Jian, Sun Le. Isoparametric element algorithm framework design based on Jaumin. Computer Aided Engineering, 2015, 24(1): 63-67(in Chinese)
  (孙运见,孙乐.基于 Jaumin 的等参单元算法框架设计.计 算机辅助工程, 2015, 24(1): 63-67)
- [7] He Xin, Zhao Zhong, Zhang Lai-Ping, et al. Research of general large scale CFD software architecture and data structure. Acta Aerodynamica Sinica, 2012, 30(5): 557-565 (in Chinese)



**ZHAO Zhong**, Ph. D., assistant professor. His research interests include unstructured grid generation, parallel computation, CFD and mesh generation software development.

**ZHANG Lai-Ping**, Ph.D., professor. His research interests include grid generation technique, numerical methods, CFD applications in aerospace industry. (赫新,赵钟,张来平.大型通用 CFD 软件体系结构与数据 结构研究.空气动力学学报,2012,30(5):557-565)

- [8] He Xin, Zhao Zhong, Zhang Lai-Ping. The research and development of structured-unstructured hybrid CFD software. Transactions of Nanjing University of Aeronautics & Astronautics, 2013, 30(Sup): 116-126
- [9] He Xin, Zhao Zhong, Zhang Lai-Ping. Validation of the structured/unstructured hybrid CFD software HyperFLOW // Proceedings of the 15th Conference on Computational Fluid Dynamics. Yantai, China, 2012: 1282-1287(in Chinese) (赫新,赵钟,张来平. 结构非结构耦合计算 CFD 软件 HyperFlow 初步验证//第 15 届全国计算流体力学会议论文 集.中国,烟台, 2012: 1282-1287)
- [10] Zhao Zhong, He Xin, Zhang Lai-ping. Numerical research of NASA high-lift Trap Wing model based on HyperFLOW. Acta Aerodynamica Sinica, 2015, 33(5): 594-602(in Chinese) (赵钟,赫新,张来平. HyperFLOW 软件数值模拟 Trap Wing 高升力外形. 空气动力学学报, 2015, 33(5): 594-602)
- [11] Schloegel K, Karypis G. Multilevel algorithms for multiconstraint graph partitioning. IEEE/ACM Conference on Supercomputing, 2005, 1900 (s1-2): 1-13
- [12] Zhang Lai-Ping, Chang Xing-Hua, Zhao Zhong, He Xin. Mesh Generation Techniques in Computational Fluid Dynamics. Beijing: Science Press, 2017(in Chinese) (张来平,常兴华,赵钟,赫新. 计算流体力学网格生成技术. 北京:科学出版社, 2017)
- [13] Jiang Qin-Gu, Jin Zhi-Yan. The hybrid MPI and OpenMP parallel scheme of GRAPES global model. Journal of Applied Meteorological Science, 2014, 25(5): 581-591(in Chinese)
   (蒋沁谷,金之雁. GRAPES 全球模式 MPI 与 OpenMP 混合 并行方案、应用气象学报, 2014, 25(5): 581-591)
- [14] He X, Zhao Z, Ma R, et al. Validation of HyperFLOW in subsonic and transonic flow. Acta Aerodynamica Sinica, 2016, 34(2): 267-275
- [15] Zhao Zhong, He Lei, et al. Large scale grid generation techniques based on global mesh refinement. CHINA, 201710655494(in Chinese)
  (赵钟,何磊等.基于全局加密的超大规模非结构网格生成 方法.专利,中国,201710655494)

**HE Lei**, master, engineer. He is interested in CFD software development and automatic testing.

**HE Xian-Yao**, master, assistant professor. His research interests include hypersonic aerodynamics and CFD software development.

**GUO Yong-Heng**, Ph. D., assistant professor. His research interests focus on theoretical and computational fluid dynamics.

**XU Qing-Xin**, Ph. D., professor. His research interest is parallel CFD development.

#### Background

This paper focuses on large scale parallel CFD software development, which is a typical and very important High Performance Computing (HPC) application. This work is supported partially by the National Key Research & Development Program of China (2016YFB020071) and the National Natural Science Foundation of China (Grant Nos. 11532016, 91530325). The aims of these programs are to develop a CFD-based software platform for aerodynamic performance

赵

simulations, multi-discipline optimization design, and virtual flight of realistic aero-vehicles. In this work, a domestic large scale parallel CFD computational framework faced to next generation, named PHengLEI, for arbitrary grids has been designed. PHengLEI is the first CFD code in the CFD community that supports not only single structured and unstructured flow solver but also coupling computation of these two kinds of solvers simultaneously.