

基于 MPI 的伪谱法大涡模拟并行计算的研究

肖红林, 罗纪生

XIAO Hong-lin, LUO Ji-sheng

天津大学 力学系, 天津 300072

Department of Mechanics, Tianjin University, Tianjin 300072, China

E-mail: x_hawk@qq.com

XIAO Hong-lin, LUO Ji-sheng. Research on parallel computing algorithm of LES used pseudo-spectral method based on MPI. Computer Engineering and Applications, 2009, 45(3): 242-244.

Abstract: Large Eddy Simulation (LES) used pseudo-spectral method achieves great accuracy and high efficiency, and parallel computing algorithm is needed because of much calculation work in condition of high Reynolds number. However, it is difficult for parallel computing algorithm with FFT to be used in practical application. Thus a new parallel computing algorithm based on Message Passing Interface (MPI) is put forward. According to the test this method is accurate, easy and stable, with higher computing speed and less computing time, which is important for LES to be widely used in engineering.

Key words: Large Eddy Simulation (LES); spectral method; Message Passing Interface (MPI); parallel computing; sub-grid model

摘要: 使用伪谱方法的大涡模拟准确、高效,但在高雷诺数情况下,计算量仍然非常巨大,需要采用并行方法,但是快速傅里叶变换的并行算法在实际应用中有很大的困难。针对这一问题,提出了一种新的基于 MPI 的伪谱法大涡模拟的并行计算方法。通过实例验证,该方法准确、易行、稳健,并且可以大幅提高计算速度,节省计算时间,这对大涡模拟在工程中的广泛应用具有重要意义。

关键词: 大涡模拟;谱方法;消息传递接口;并行计算;亚格子模型

DOI: 10.3778/j.issn.1002-8331.2009.03.073 **文章编号:** 1002-8331(2009)03-0242-03 **文献标识码:** A **中图分类号:** O246

大涡模拟(Large Eddy Simulation, LES)方法目前是数值研究湍流运动的有效工具之一,该方法有很好的工程应用前景,在气象、燃烧等领域得到广泛应用。快速傅里叶变换的产生推进了谱方法在数值计算中的应用,在大涡模拟中使用谱方法会使数值运算更准确、高效。然而,由于湍流运动的复杂性,使得数值计算在大雷诺数情况下的工作量非常巨大,在一定程度上限制了大涡模拟在工程实际中的应用。

计算流体力学工作者们努力寻求更好的办法来大规模提高计算速度,并行计算在此背景下应运而生。要实现谱方法大涡模拟的并行,首先要解决快速傅里叶变换的并行算法。国外围绕快速傅里叶变换的并行计算进行了多项研究和开发,国内从 20 世纪 80 年代初开始也开展了并行算法研究。不过这些算法有如下不足之处^[1]:(1)大多是基于特定的体系结构如碟形网络、超立方体计算机,这些高档计算机虽然具有较高的性能,但是其昂贵的价格并不是普通的用户所能接受的;(2)这些算法多是假设任务间的通信时延为 0、通信无冲突、并行处理器的完全互连和可用处理器的个数不受限制等。这些假设在实际应用系统中不一定成立。

既然使用快速傅里叶变换的并行算法在实际中有很大的

困难,那么是否有其它的办法呢?

鉴于这一情况,针对硬件的具体情况和谱方法大涡模拟问题本身的特点,放弃使用快速傅里叶变换的并行算法这条路。而是,在 MPI 并行环境下,使用主从模式,采用对大涡模拟方程中不同项在不同方向进行并行的方法,对槽道湍流进行数值模拟。并将并行结果与串行结果进行对比,结果一致。

1 MPI 并行简介

MPI(Message Passing Interface)是一个消息传递接口的标准^[2],用于开发基于消息传递的并行程序。MPI 子集由 6 个基本函数组成:MPI 初始化、MPI 结束、获取进程编号、获取指定通信域的进程数、消息发送和消息接受。MPI 支持以下四种通信模式:标准通信模式、缓冲通信模式、同步通信模式、就绪通信模式。MPI 提供了与 C 和 Fortran 语言的绑定。

2 本文问题的描述

考虑平面槽道流,设流动的几何条件和坐标系如图 1 所示,其中 x 为流向, y 为法向, z 为展向。

对不可压缩流动的连续方程和 Navier-Stokes 方程进行滤

基金项目: 国家自然科学基金(the National Natural Science Foundation of China under Grant No.10232020)。

作者简介: 肖红林(1978-),男,博士研究生,主要研究方向:湍流、流动稳定性、并行计算;罗纪生(1954-),男,教授,博士生导师,主要研究方向:流动稳定性及湍流。

收稿日期: 2008-07-07 **修回日期:** 2008-08-07

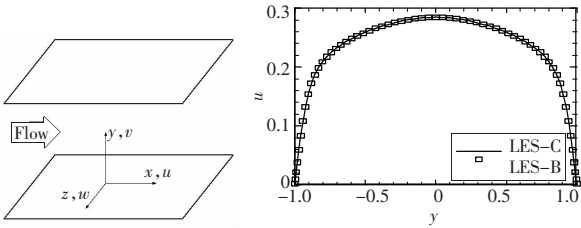


图1 平面槽道流的示意图

图2 平均速度的对比

波, 采用涡粘性假设, 得到大涡模拟的控制方程:

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0 \quad (1)$$

$$\frac{\partial \bar{u}_i}{\partial t} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} = -\frac{\partial \bar{P}}{\partial x_i} + \frac{1}{Re} \frac{\partial}{\partial x_j} \frac{\partial \bar{u}_i}{\partial x_j} + \frac{1}{Re} \frac{\partial}{\partial x_j} [\nu_t (\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i})] \quad (2)$$

其中, $Re = U_m h / \nu$, U_m 是最大平均速度, h 为半槽宽度, ν 为运动学粘性系数, ν_t 为涡粘性系数。

亚格子模型采用文献[3]建议的模型: $\nu_t = (c \Delta \Delta)^{2/3} |S|$ 。其中, c 是一常数, 一般取为 0.1; $\Delta = (\Delta x \Delta y \Delta z)^{1/3}$, 其中 Δx 、 Δy 和 Δz 分别是在 x 、 y 和 z 方向上的网格宽度; 局部应变率定义为 $S = (2S_y S_y)^{1/2}$, 其中 $S_y = (\partial u_j / \partial x_i + \partial u_i / \partial x_j) / 2$; D 为修正函数, 形式如下:

$$D = D_1 \times D_2$$

$$D_1 = 1 - \exp[-(y^+ / A^+)^3]$$

$$D_2 = c_2 \{1 - th[c_1 (y^+ - y_0^+)]\} / 2 + 1$$

其中, $A^+ = 25$, $y_0^+ = 80$, $c_1 = 32$, $c_2 = 200$ 。详细情况可参该文献[3]。计算时在空间采用伪谱方法, 即在流向和展向做傅里叶展开, 在法向用切比雪夫多项式展开。对时间离散时, 线性项采用二阶精度的隐格式, 非线性项采用二阶精度的显格式。设采用的计算网格数沿流向、法向、展向为 $m \times l \times n$, 则得到 $m \times n$ 个形如:

$$Ax^{t+\Delta t} = Bx^t + Q^t + T^t \quad (3)$$

的方程组, 方程的详细推导见文献[4]。其中, t 为当前时刻, $t+\Delta t$ 为下一时刻; A 、 B 为 $l \times l$ 阶的系数矩阵; x^t 为当前时刻已求得的速度场; $x^{t+\Delta t}$ 为下一时刻将要求的速度场; Q^t 为非线性项, 它由当前时刻速度场求得; T^t 为大涡项, 它也由当前时刻速度场求得。对于大涡模拟的每一次循环迭代, 主要由三部分组成: 求非线性项 Q^t 、求大涡项 T^t 、解方程 $Ax^{t+\Delta t} = f^t$ (其中 $f^t = Bx^t + Q^t + T^t$)。

在求方程组(3)中的非线性项 Q^t 和大涡项 T^t 时要进行非线性乘法, 为了不产生混淆误差, 需要将相关量的阶数扩大一倍再变到物理空间进行乘法, 然后再变回谱空间, 并将阶数缩回到原来阶数。由于需要在流向和展向都要扩大一倍, 而且在计算大涡项时要进行多次非线性乘, 致使计算量非常大, 需要进行并行计算。但由于傅里叶变化需要全场信息, 如果在流向和展向进行并行, 将会带来大量的信息传递, 耗费大量的机时。所以, 该方法是在法向进行并行。

方程组(3)是在法向配置点上列出的, 解方程是在法向进行的, 所以本文的方法是在展向(也可以在流向)进行并行。

3 并行算法

由于解方程和求非线性项及大涡项不是在同一方向进行的, 每迭代一次需要数据汇总一下, 所以并行采用了主从模式。主进程要进行的工作为: 读入初始数据、进行法向切比雪夫变化、收集和发送数据、输出数据。从进程要进行的工作为: 求非线性项 Q^t 和大涡项 T^t 、解方程 $Ax^{t+\Delta t} = f^t$ (其中 $f^t = Bx^t + Q^t + T^t$)。

求非线性项和大涡项时沿法向进行并行, 将流场沿法向分割, 如图3所示; 解方程时沿展向进行并行(也可以沿流向), 将流场沿展向(也可以沿流向)分割, 如图4所示。

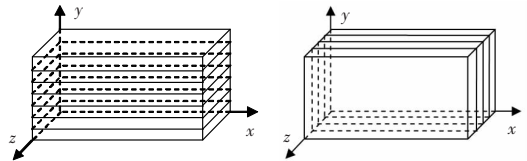


图3 沿法向分割流场

图4 沿展向分割流场

假设有 p 个从进程, 设为 $P_i, i=1, 2, \dots, p$, 设主进程为 P_0 。

初始流场 x^0 及每次循环迭代得到的速度场 x^t 三个方向均在谱空间, 则并行的流程图如图5所示, 其中:

- (1) 主进程 P_0 读入初始流场 x^0 , 并将初始流场 x^0 沿展向分割成 p 份, 分割后的流场设为 $x_{zi}^0, i=1, 2, \dots, p$, 然后将 x_{zi}^0 发给从进程 P_i ;
- (2) 从进程 P_i 接收来自主进程 P_0 的初始流场 $x_{zi}^0, i=1, 2, \dots, p$;
- (3) 主进程 P_0 求速度场 x^t 在法向的导数 dx^t/dy , 并将速度场 x^t 及其导数 dx^t/dy 在法向变到物理空间, 将变化后的 x^t 和

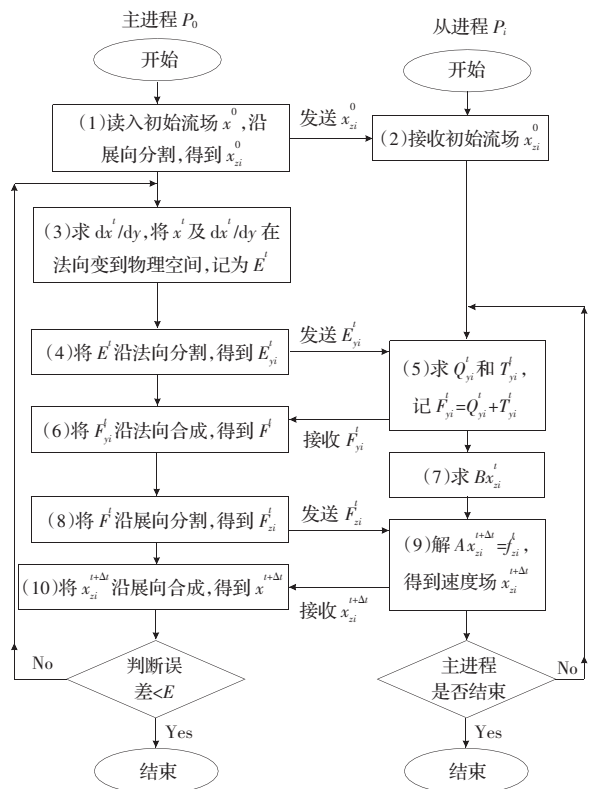


图5 并行程序流程图

dx^i/dy 记为 E^i ;

(4) 主进程 P_0 将 E^i 沿法向分割成 p 份, 设为 $E_{y_i}^i, i=1, 2, \dots, p$, 将 $E_{y_i}^i$ 分别发送给 p 个从进程 $P_i, i=1, 2, \dots, p$;

(5) 从进程 P_i 收到数据 $E_{y_i}^i$ 后, 进行求非线性项 $Q_{y_i}^i$ 和大涡项 $T_{y_i}^i$ 的计算, 包括多次的沿流向和展向的求导、傅里叶变换、非线性乘法等, 将得到的结果, 记为 $F_{y_i}^i (F_{y_i}^i = Q_{y_i}^i + T_{y_i}^i)$, 然后将 $F_{y_i}^i$ 向主进程 P_0 发送;

(6) 主进程 P_0 接收各从进程 P_i 的数据 $F_{y_i}^i$ 后, 将 $F_{y_i}^i$ 沿法向合成, 记为 F^i ;

(7) 与步骤(6)同时, 各从进程 P_i 进行求 $Bx_{z_i}^i$ 的计算, $i=1, 2, \dots, p$;

(8) 主进程 P_0 将 F^i 沿展向(也可沿流向)分割成 p 份, 分割后的 F^i 记为 $F_{z_i}^i, i=1, 2, \dots, p$, 将 $F_{z_i}^i$ 分别发送给从进程 P_i ;

(9) 从进程 P_i 接收数据 $F_{z_i}^i$, 到此, 方程(3)的右端项均已求得 ($f_{z_i}^i = Bx_{z_i}^i + F_{z_i}^i = Bx_{z_i}^i + Q_{z_i}^i + T_{z_i}^i$), 然后进行解方程 $Ax_{z_i}^{i+\Delta t} = f_{z_i}^i$ 的计算, 得到速度场 $x_{z_i}^{i+\Delta t}$ (三个方向均在谱空间), 然后将 $x_{z_i}^{i+\Delta t}$ 发送给主进程 P_0 ;

(10) 主进程 P_0 收集各从进程 P_i 的数据 $x_{z_i}^{i+\Delta t}$ 后, 沿展向合成, 得到三个方向均在谱空间的速度场 $x^{i+\Delta t}$ 的全场信息。

4 结果对比

计算域在 x, y, z 方向的长度分别为 $6.7\pi, 2.5\pi$, 计算网格数为 $64 \times 65 \times 64$, 其中 x, z 方向为均匀网格, y 方向按切比雪夫配置点分布。采用 Fortran90 语言, 分别编写了串行、并行程序, 在曙光 TC1700 集群服务器下计算, 系统为 Linux, 网络为千兆网。

图 2 给出了并行和串行两种算法下平均流速剖面的对比; 图 6 给出了并行和串行两种算法下雷诺应力的对比; 图 7 给出了并行和串行两种算法下扰动速度均方根的对比。图中横

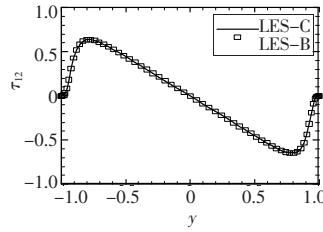


图6 雷诺应力的对比

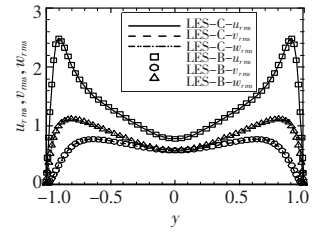


图7 扰动速度均方根的对比

坐标为法向坐标, 曲线为串行结果, 符号为并行结果。从图中可以看到, 并行情况下的结果与串行情况下的结果完全一致。

本算例并行加速比为: $S = T_c / T_p = 9.64$, 其中: T_c 表示串行运行时间, 即指在串行计算机上程序从开始到结束所用时间; T_p 表示并行运行时间, 即指从并行计算开始时刻到最后的处理器完成运算所经过的时间。并行效率 E 为: $E = S/p \times 100\% = 68.9\%$, 其中: p 表示参与并行计算的 CPU 个数, 本文 $p=14$ 。

5 结论

提出了一种新的基于 MPI 的伪谱法大涡模拟的并行计算方法, 该方法使用主从模式, 对大涡模拟方程中的不同项在不同方向并行, 该方法直观、易行、稳健。用该方法得到的并行结果与串行结果进行了对比, 结果完全一致。该并行方法能大幅提高计算速度, 节省计算时间, 解决了谱方法大涡模拟并行计算在实际应用中遇到的困难。这将有助于在工程实际中更广泛的使用大涡模拟方法, 也有助于大涡模拟方法在高雷诺数下理论研究中的应用。

参考文献:

- [1] 于秀敏. 快速傅立叶变化并行算法概述[J]. 中国科技信息, 2005, 9: 52-52.
- [2] 都志辉. 高性能计算并行编程技术-MPI 并行程序设计[M]. 北京: 清华大学出版社, 2001.
- [3] 肖红林, 罗纪生. 大涡模拟中亚格子模型的改进及其在槽道湍流中的应用[J]. 航空动力学报, 2007, 22(4): 583-587.
- [4] 肖红林, 周恒. 槽道湍流的大涡模拟[D]. 天津: 天津大学, 2004.
- [5] 王盛卫. 集成楼宇控制系统辅助之变风量空调系统的实时优化控制[C]// 全国暖通空调制冷 1998 年学术年会学术文集. 北京: 中国建筑工业出版社, 1998: 378-386.
- [6] 李克欣. 变风量空调系统的 VPT 控制法及其应用[J]. 暖通空调, 1999, 29(3).
- [7] 邱步. 变风量空调系统的设计和工程实例[J]. 现代空调, 1999, 12.
- [8] Yanagawa S, Miki I. PID auto-tuning controller using a single neuron for DC servomotor[C]// IEEE Int Sym on IE, Xi'an, China, 1992, 5.
- [9] 王宁, 涂建, 陈锦江. 使用单个自适应神经元的智能控制[J]. 华中理工大学学报, 1993, 21(3).
- [10] 王晓东. 直流传动系统中单神经元自适应智能控制的实现[J]. 机电传动, 1996, 5.
- [11] 丁红, 舒迪前. 无需辨识模型的 PSD 自适应控制器及其应用[J]. 冶金自动化, 1994, 18(2).
- [12] 李明. 自适应神经元非模型多变量系统解耦控制[J]. 计算机仿真, 2003.
- [13] 董阿妮. 变风量空调系统自适应神经元解耦控制与变静压协调控制[D]. 西安: 西安建筑科技大学, 2004.

(上接 197 页)