

基于 Matlab 分布式工具箱的流场计算及其可视化

蔡 群¹, 周美莲², 段杰峰¹, 李 青¹

(1. 上海大学 计算机工程与科学学院, 上海 200072;

2. 上海大学 机电工程与自动化学院, 上海 200072)

摘 要: 讨论了格子 Boltzmann 方法 LBGK D2Q9 模型的计算过程和计算特点。通过该模型模拟了一类典型的流体绕流问题的计算及其可视化, 研究并实现了利用 Matlab 分布式计算工具箱进行分布式并行处理的问题。讨论了利用虚拟目录及文件的方式传递数据的过程, 实现了跟踪方式的 LBGK 流体计算模型科学计算可视化系统。

关键词: 分布式计算; 格子模型方法; 科学计算可视化

中图分类号: TP311

文献标识码: A

文章编号: 1673-629X(2007)09-0051-04

Flow and Its Computational Visualization Based on Matlab MDCT

CAI Qun¹, ZHOU Mei-lian², DUAN Jie-feng¹, LI Qing¹

(1. School of Computer Engineering and Science, Shanghai University, Shanghai 200072, China;

2. College of Mechatronics Engineering and Automation, Shanghai University, Shanghai 200072, China)

Abstract: The courses and characteristics of the computation based on the model D2Q9 of lattice Boltzmann method was discussed in this paper. With this model, the computing and visualization of one typical flow question were simulated, and at the same time, doing distributed parallel processing was researched and realized with Matlab distributed computing toolbox. The courses of transfer data using virtual directory and file system were discussed also, moreover, a visualization in scientific computation system which is based on tracking method was realized.

Key words: distributed computing; lattice Boltzmann method; science computing visualization

0 引 言

格子 Boltzmann 方法 (Lattice Boltzmann Method, LBM) 作为一种流场计算方法在复杂流动的数值模拟中越来越受到重视。特别是 20 世纪 90 年代后兴起的 LBGK (Lattice Bhatnagar - Gross - Krook) 模型更以其编程简单、边界处理简便、无统计噪声等优点而得到广泛的应用。科学计算的目的不仅仅是为了获取数值, 更重要的是洞察复杂现象的形成历程及其机理。复杂流动的计算可视化将利用图形图像、动画等手段表现复杂数据^[1]。

LBGK 模型的计算中, 主要的是一个松弛迭代过程。该迭代过程的每一步均计算出流场的密度、速度等物理量形成宏观流场。可以用跟踪方式进行流场计算的可视化。文中用二维 9 速模型 D2Q9 为例, 利用 Matlab 分布式计算工具箱实现了计算与可视化的并行

处理。

1 LBGK D2Q9 模型

流体力学是研究流体运动规律的一门学科。数学上, 流体的宏观运动可以用一组非线性偏微分方程描述, 即 Navier - Stokes 方程。但是由于流体运动的复杂性, 除了某些简单的情况外, 很难获得这些偏微分方程的精确解。对于大多数的实际问题, 必须采用实验方法或数值方法。随着计算机技术的发展, 数值模拟方法已经成为流体力学研究中的一种重要手段和工具, 并且日益受到人们的重视。这一领域已经发展成为流体力学的一个分支, 即计算流体力学 (Computational Fluid Dynamics, CFD)。

LBGK 是一种模拟流体流动的数值模型方法, 它将流体看成许多粒子微团组成, 这些粒子微团可以向空间的若干方向任意流动。最早的 LBGK 模型是由 Qian^[2]、Chen^[3] 等人于 1992 年分别独立提出的一种单松弛碰撞模型, 其主要思想就是以简单的松弛过程, 刻画粒子在短时间内的局部运动规律, 经过反复演化逐

收稿日期: 2006-12-07

基金项目: 国家自然科学基金资助项目 (60373071)

作者简介: 蔡 群 (1982-), 女, 江苏泰兴人, 硕士研究生, 研究方向为高性能计算、计算可视化。

步形成复杂多变的宏观现象, LBGK 模型已经应用到许多问题的模拟^[4,5]。它主要有以下优点:

- (1) 边界条件容易处理;
- (2) 算法简单, 程序容易实现;
- (3) 并行程度高;
- (4) 模型的宏观行为符合所研究的宏观方程。

1.1 D2Q9 模型

随着对格子 Boltzmann 方法的研究的不断深入, 越来越多的学者开始提出一些各自不同的格子 Boltzmann 模型来模拟复杂的流体力学问题。其中, Qian 等人于 1992 年提出的 DnQb 模型^[2](D 是 dimension 的缩写, n 代表维数, b 代表离散速度方向), 至今仍然是应用的最为广泛。

D2Q9 模型划分的是正方形网格, 每个格点和周围的八个格点相邻。把时间离散成时间步, 在每个时间步, 粒子只能停留在格点上或向相邻格点运动。因此, 在每个格点上有三种类型的粒子, 分别为静止粒子(其速度为零)、速度为 1 的粒子和速度为 $\sqrt{2}$ 的粒子, 其中速度为 1 的粒子沿水平或垂直方向运动, 速度为 $\sqrt{2}$ 的粒子沿对角线方向运动。图 1 表示 D2Q9 模型的网格划分和速度方向。

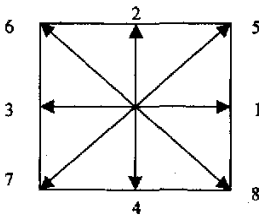


图 1 D2Q9 模型的网格划分和速度方向

在二维正方形网格点 $x = (x_1, x_2)$ 处, t 时刻, 流体微团密度按 9 个离散速度方向 $e_i (i = 0, 1, \dots, 8)$ 的分布函数记为 $f_i(x, t)$, 其演化方程为

$$f_i(x + ce_i\Delta t, t + \Delta t) = (1 - \omega)f_i(x, t) + \omega f_i^{\text{eq}}(x, t) \quad i = 0, 1, 2, \dots, 8 \quad (1)$$

其中离散速度集

$$e_i = \begin{cases} (0, 0) & i = 0 \\ (\cos(i-1)\pi/2, \sin(i-1)\pi/2) & i = 1, 2, 3, 4 \\ \sqrt{2}(\cos((i-1)\pi/2 + \pi/4), \sin((i-1)\pi/2 + \pi/4)) & i = 5, 6, 7, 8 \end{cases}$$

$c = \Delta x / \Delta t$, 声速 $c_s = c / \sqrt{3}$, 流体粘性系数 ν 与松弛因子 ω 的关系为 $\nu = c_s^2(1/\omega - 1/2)$, 平衡态分布函数

$$f_i^{\text{eq}}(x, t) = b_0 \rho \left[1 + \frac{e_i \cdot u}{c_s^2} + \frac{(e_i \cdot u)^2}{2c_s^2} + \frac{u^2}{2c_s^2} \right] \quad (2)$$

其中系数 $b_0 = 4/9, b_1 = b_2 = b_3 = b_4 = 1/9, b_5 = b_6 = b_7 = b_8 = 1/36$, 宏观密度和动量

$$\rho(x, t) = \sum_i f_i(x, t),$$

$$\rho(x, t)u(x, t) = \sum_i f_i(x, t)e_i \quad (3)$$

上述模型在一定的条件下能够逼近模拟流体运动的 Navier-Stokes 方程。

1.2 D2Q9 模型的计算

初始化: 对于给定的物理参数和宏观的初始条件及边界条件, 计算 D2Q9 模型的参数, 令初始的密度分布函数 $f_i(x, 0) = f_i^{\text{eq}}(x, 0)$;

演化过程: 粒子微团在每个时间步的运动又可分为“松弛”(由公式(3)、(2)计算(1)的右端项)、“流动”(将“松弛”步的结果, 按其速度方向移动到最邻近的网格点上);

反复演化, 直到流动稳定或者计算到指定的时刻为止。

图 2 中, 圆圈中的数字代表时间步, 整个圆圈表示该时间步的宏观物理场(如: 速度场、密度场等)。显然, 上述计算过程中, 每一演化过程均可得到复杂流动的宏观密度场、速度场; 然后, 通过颜色映射、绘制等步骤, 可实现复杂流动的可视化处理。

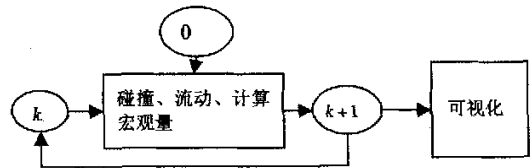


图 2 LBGK 模型的计算及其可视化

2 Matlab 可视化及分布式计算工具箱

Matlab 是由美国 MathWorks 公司开发推出的, 是当今国际科学界最具有影响力, 也是最有活力的软件之一。它起源于矩阵运算, 并已发展成一种高度集成的计算机语言。它提供了强大的科学运算, 灵活的程序设计流程, 高质量的图形可视化与界面设计, 具有便捷的与其他程序和语言接口的功能。Matlab 还提供了功能强大的工具箱函数, 并允许用户构造自己的工具箱函数, 很容易由用户自行扩展。此外, Matlab 提供了强大的可视化能力, 可以实现二维、三维以及四维作图能力, 高级的可视化工具包括表面着色、光照、图像显示等。利用 Matlab 可以方便地实现数据的可视化显示^[6]。

2.1 Matlab 可视化

Matlab 提供了强大的图形表达功能, 可以用简单的命令方便地绘制二维、三维图形, 尤其是在计算流体力学中, 经常涉及到等值线以及速度、压力的分布, 速度的矢量合成图等, 这些在 Matlab 中是非常简单而又实用的命令。文中主要用到的绘图命令有曲线图和向

量图。函数 `plot(Xn, Yn, LineSpec, 'PropertyName', PropertyValue, ...)` 可以绘制二维曲线图。其中, X_n 或 Y_n 为矩阵, 该向量与矩阵的行或列配对进行绘图, 取行还是取列决定于向量的行或列的维数是否与矩阵匹配; `LineSpec` 指定直线的线型、标记和颜色; 'PropertyName', PropertyValue 为所有由 `plot` 函数创建的图形对象设置属性值。 `quiver(X, Y, U, V, scale, LineSpec, 'filled'...)` 函数可以绘制向量图, 用箭头显示图中各点处的向量大小和方向。其中, 箭头指示的方向为向量的方向, 箭头的长短表示向量的大小。此函数对每个 X 和 Y 配对数据绘向量图。 X 和 Y 为向量, `length(X)` 且 `length(Y) = m`, 其中 `[m, n] = size(U) = size(V)`。向量 X 对应于 U 和 V 的列, 向量 Y 对应于 U 和 V 的行; `scale` 对向量自动设置显示比例, 如果 `scale = 2`, 则将原向量的长度乘以 2 以后显示。若 `scale = 0`, 则不自动设置显示比例; `LineSpec` 参数指定线型、标注和颜色。 `quiver` 函数在向量原点处标记; `filled` 填充由 `LineSpec` 指定的标记。

2.2 Matlab 分布式计算工具箱

Matlab 在科学计算中扮演了重要的角色, 被大量的工程技术人员使用。然而传统 Matlab 产品的局限在于 Matlab 程序都是串行的。虽然 Matlab 拥有大量高效的代码, 然而不支持并行计算大大地限制了 Matlab 在工程计算领域的应用。

针对这一问题, MathWorks 公司推出了 Matlab 分布式计算引擎 (Matlab Distributed Computing Engine, MDCE) 和工具箱 (Matlab Distributed Computing Toolbox, MDCT) 可以在机群环境下并行执行多个任务。如图 3 所示。一个 Job 可以被分割成若干个彼此独立又相互联系的 task。Job 及其 tasks 是在客户机群 (Client Session) 中定义的。客户机 (Client) 使用 MDCT 来进行 Jobs 和 tasks 的定义; MDCE 通过评估某个 Job 的 tasks 并返回其计算结果来执行这个 Job。Job manager 是引擎的一部分, 它协调 Jobs 的执行和 tasks 的评估。Job manager 把 tasks 发布在引擎的独立的 Matlab Sessions 即 workers 上。想要在 workers 上发布 tasks, 除了使用 Job manager 还可以选择第三方的 scheduler, 比如 LSF。与在单个结点上执行相同的算法相比较, 把计算任务分配到各个结上去执行大大地节省了时间提高了效率。

3 流场可视化系统框架设计与实现

文中按计算与可视化将整个处理过程分为两个部

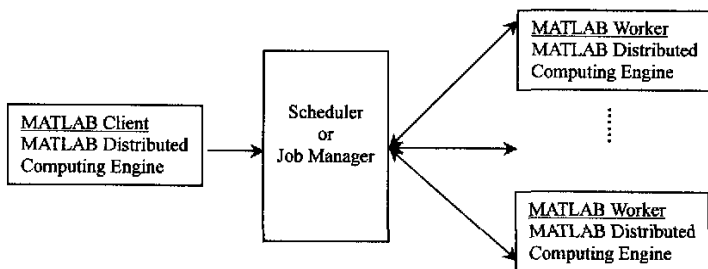


图3 基本的分布式计算配置

分, 它们通过文件的形式传递数据, 由不同的 Worker 进行并行处理。

首先配置网络环境, 将存放计算结果的子目录映射成虚拟目录, 如 `//Cherry/mymatlab`。主要的 Matlab 代码如下:

```
manager = findResource('scheduler', 'type', 'jobmanager');
fig = setup_lbgk();
job = createJob(manager);
p = '\\Cherry\\mymatlab';
set(job, 'PathDependencies', {p});
t1 = createTask(job, @Computing, 0, {});
t2 = createTask(job, @task_lbgk, 0, {}, ...
    'RunningFcn', @taskRun_lbgk, ...
    'UserData', fig);
submit(job);
waitForState(job, 'finished');
```

其中 `findResource('scheduler', 'type', 'jobmanager')` 获得一个 `jobmanager` 对象 `manager`, 其后由 `manager` 创建一个 `job`。 `fig = setup_lbgk()` 调用 `m` 文件 `setup_lbgk.m` 生成一个图形对象 `fig`。 `p = '\\Cherry\\mymatlab'` 及 `set(job, 'PathDependencies', {p})` 在此 `job` 中开放存放计算结果的文件的读写权限。 `t1 = createTask(job, @Computing, 0, {})` 创建进行 LBGK 模型计算的任务, `Computing.m` 内包含一个 `mex` 文件 `LBGKComputing.mexw32`, 是对 `LBGKComputing.cpp` 进行 `mex` 编译的结果。 `t2 = createTask(job, @task_lbgk, 0, {}, 'RunningFcn', @taskRun_lbgk, 'UserData', fig)` 调用 `taskRun_lbgk.m` 创建可视化任务, 它读取数据并在 `fig` 中显示图形。最后用 `submit(job)` 递交任务, 并用 `waitForState(job, 'finished')` 等待任务的完成。

4 算例

本算例是一个典型的复杂流动问题——圆柱绕流。流场划分成 140×70 的网格点, 在一定的初始条件下, 按 LBGK 模型的计算流程执行。图 4 是执行 1000 步计算时的情形, 其中箭头表示各处的速度场, 颜色的深浅表示流场的密度。

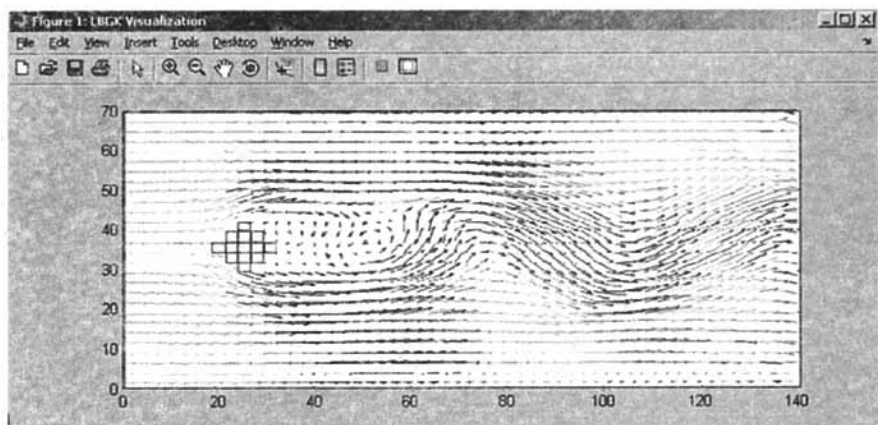


图 4 流场可视化结果

5 结 论

讨论了 LBGK D2Q9 模型在 Matlab 的分布式计算工具箱下计算与可视化的分布式并行处理方法及实现技术。实现了该模型的跟踪方式的可视化处理。利用 Matlab 的强大的科学运算和工具箱函数,便捷的接口的功能,高质量的图形可视化与界面设计,实现了 CFD 数值模拟中流场计算和可视化显示,对提高程序的运行效率、运算结果的分析效率以及改善分析计算的工作环境有着积极的意义。

参考文献:

- [1] 李晓梅,黄朝晖. 科学计算可视化导论[M]. 长沙: 国防科

技大学出版社,1996.

- [2] Qian Y, d'Humieres D, Lallemand P. Lattice BGK models for Navier - Stokes equation[J]. Europhys Lett, 1992, 17: 479 - 484.
 [3] Chen H, Chen S, Matthacus W H. Recovery of the Navier - Stokes equations using a lattice gas Boltzmann method[J]. Phys Rev A, 1992, 45: 5339 - 5342.
 [4] Li Q, Zheng C, Wang N, et al. LBGK simulations of Turing patterns in CIMA model[J]. J Scientific Computing, 2001, 16 (2): 121 - 134.
 [5] 郭照立,郑楚光,李 青,等. 流体动力学的格子 Boltzmann 方法[M]. 武汉: 湖北科学技术出版社, 2002.
 [6] 张志勇. 精通 MATLAB5. 3[M]. 北京: 北京航空航天大学出版社, 2000.

(上接第 50 页)

(6) 减后数据集的频繁项集为 L_{D-d} 。

4 结束语

文中在对正负增量式关联规则更新技术^[4-6]深入研究基础上,提出了相应的算法。通过分析可以看出: 整个更新过程只扫描一次数据集;通过连接项集产生候选项,增强了产生候选项的针对性和有效性,提高了候选项的支持事务计数的效率;充分利用已知挖掘结果和正在挖掘过程中得到的信息,竭力获得无需计算支持数便已知的 L_{D-d} 中的项目,剪枝过程贯穿于整个算法,在算法的运行时耗上具有很明显的优势。历史数据集越大,剪枝效果越加明显,越能显现出本算法的优越性。由于本算法经常要判断一个项目集是否存在于另一个项目集中,可以在此问题上做一定的研究(如可以利用项集中项目有序的特点等),以进一步提高本算法的效率。

参考文献:

- [1] Han Jiawei, Kamber M. 数据挖掘概念与技术[M]. 北京: 机械工业出版社, 2001.
 [2] DAVID W, CHEUNG J, NGVT, et al. Maintenance of Discovered Association Rules in Large Database: An Incremental Updating Technique[C]//In Proc 12th Int. Conf. on Data Engineering. New Orleans, Louisiana: IEEE Computer Society, 1996: 106 - 114.
 [3] DAVID W, CHEUNG J, LEE S D, et al. A general Incremental Technique for Maintaining Discovered Association Rules [C]//In Proceedings of the Fifth International Conference on Database Systems for Advanced Applications. Melbourne, Australia: [s. n.], 1997: 185 - 194.
 [4] 陈劲松,施小英. 一种关联规则增量更新算法[J]. 计算机工程, 2002, 28(7): 106 - 107.
 [5] 冯玉才,冯剑琳. 关联规则的增量式更新算法[J]. 软件学报, 1998, 9(4): 301 - 306.
 [6] 孙 浩,赵 霁. 一种关联规则增量更新算法[J]. 系统工程与电子技术, 2004, 26(5): 676 - 677.