

三维流体动力学并行软件包 NaSt3DGP 及其应用

常 英 立

(上海水产大学, 上海 200090)

提 要

该文介绍了解算直角坐标系下不可压缩 Navier-Stokes 方程的大型有限差分并行软件包 NaSt3DGP。介绍了其采用的区域分解算法、差分格式及计算流程;对并行效率进行了测试,作出了处理器数目和计算效率的关系曲线;把此程序应用到旋转的行星流体动力学中,分别用自检验和互检验的方法证明了修改后程序的可靠性,并且得到了满意的非线性数值模拟结果。

主题词: 流体动力学 — 软件包 — 区域分解 — 有限差分 — 数值模拟

分类号: TP317, O351.2

1 引 言

大型有限差分并行软件包 NaSt3DGP 由德国波恩大学 Michael Griebel、Roberto Croce、Frank Koster 和 Michael Meyer 用 C++ 语言编写而成,主要用于解算三维直角坐标系下不可压缩的 Navier-Stokes 方程。这是一个开源软件,很容易进行修改,以便应用于求解更复杂的流体动力学问题。例如,在行星流体动力学研究中,应用此软件可以得到行星流场的非线性数值解^[1]。

2 程序的实现及使用方法

该程序求解的方程为:

$$\frac{\partial u}{\partial t} + u \cdot \nabla u = \frac{g}{Fr} - \nabla p + \frac{1}{Re} \nabla^2 u \quad (1)$$

$$\nabla \cdot u = 0 \quad (2)$$

$$\left(\frac{\partial \Theta}{\partial t} + u \cdot \nabla \Theta \right) = \frac{1}{RePr} \nabla^2 \Theta \quad (3)$$

式(1)是速度场方程。式中 u 表示流体的运动速度; g 表示重力加速度; Fr 称为弗劳德 (Froude) 数,其物理意义是表示惯性力与重力之比,流体动力相似的流动其 Fr 值相同; Re 称为雷诺 (Reynolds) 数,其物理意义为惯性力与粘性力的比值。两种流动的粘性力作用相似,它

们的雷诺数必定相等。式(2)表示不可压缩流的连续性方程。式(3)表示温度(或能量)方程。式中 Θ 表示热力学温度,无量纲 Pr 称为普朗特(Prandtl)数,它表征了流体中动量扩散与热量扩散能力的比值。此程序可以求解出任意时刻的速度场和温度场。

NaSt3DGP 采用的是并行算法。并行计算的原理是:整个计算区域分割成多个子区域,然后分配给多个处理器同时计算。子区域的初始信息和几何信息分别装载入各子区域对应处理器的内存中,在每一个处理器中启动计算进程,由主进程调度各处理器的计算工作。在每一次全场的扫描过程中,由各处理器完成子区域的计算并在边界完成数据交换。

对非均匀网格的空间离散化采用了有限差分的格式,该方法是直接将微分问题变为代数问题的近似数值解法。其数学概念直观,表达简单,是发展较早且比较成熟的数值方法。

2.1 并行的实现

并行的实现方法有很多种,例如区域分解并行(Domain Decomposition Method, DDM)、矩阵运算并行(主要是将方程归结为大规模矩阵的并行求解)和上述两种方法的综合并行等。

区域分解并行在进行区域分解时就考虑了实际的物理效应,所以其并行效率是最高的,而且各子域的计算可以独立进行,在不同的子区域可以采用不同的算法^[2]。

矩阵运算是把实际问题通过一系列的模型转化,归结为大型稀疏矩阵的求解。对于此类问题,一般有通用的并行解算包。

2.1.1 本程序的并行算法:

本程序基于区域分解法,并在 MPI 上实现。在整个运行过程中,仅执行了一次区域分解,此后所有流场的运算全部在子区域中进行。数据交换只限于相邻区域的网格上,整个过程十分简洁高效。流场区域为长方体,分解策略为将流场分解为若干个小长方体,参见图 1。

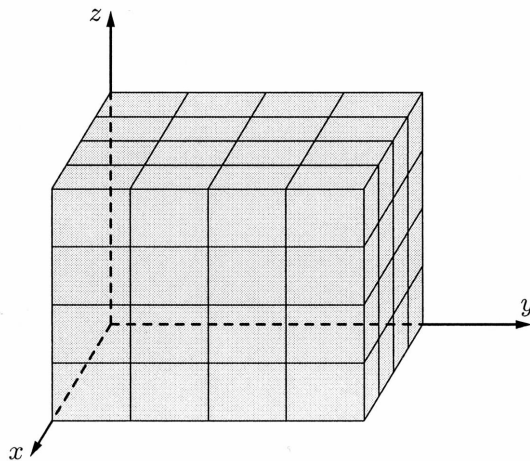


图 1 流场分解示意图

Fig. 1 Flow decomposition diagram

2.1.2 并行效率测试

在不考虑通讯延迟的情况下当计算使用的处理器数目越多,计算速度会越快,但实际上并不可能如此。为此我们在 $60 \times 60 \times 60$ 的网格数下测试了使用不同处理器数时的运算速度。测试结果见表 1。

表 1 处理器数与运算速率关系表

Table 1 The relationship of processors numbers and speeds of Operational

处理器数目	4	6	8	12	18	27
每步计算时间/s	0.0461	0.0317	0.0252	0.0191	0.0162	0.0136
处理器数目	32	42	48	64	80	
每步计算时间/s	0.0135	0.0134	0.0133	0.0140	0.0145	

从表中我们容易看到,运算速度并不随参加运算的处理器数的增多而单调上升。这是因为计算速度不仅和处理器数目有关,还和交换信息量有关。在网格点数目确定的情况下,当分配给程序的处理器增加的时候,运算能力也会随之增强,但是同时处理器之间需要交换的信息也越来越多,信息交换所需要的时间也越多。起初交换信息的增长速度远小于运算能力的增强速度,因此表现为计算速度明显加快;但不久以后随着需要交换的信息的增长,抵消了运算能力的增强,这时表现为计算速度不再明显加快;随着交换信息进一步增多,最终将导致计算速度开始下降。在图 2 中,我们可以更加清晰地看到这一过程。所以,在进行并行计算之前,必须进行并行效率测试。以便找到一个合适的处理器数目来获得最快的运算速度,以提高并行运算的效率。

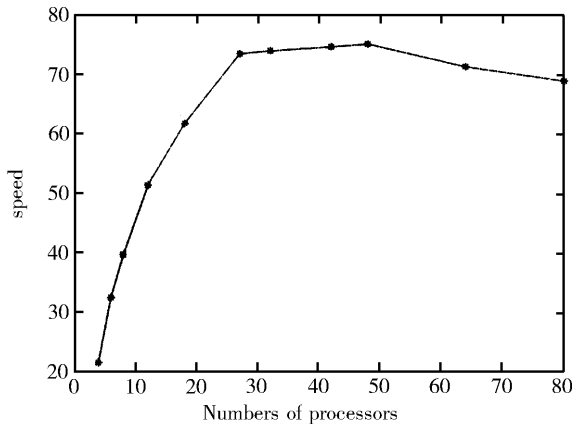


图 2 某网格下处理器数与运算速率的关系

Fig. 2 The relationship of processors numbers and speed of Operational at the certain grids

2.2 时间差分格式的实现

该程序解算步骤如下:

$$\frac{\tilde{u}^n - u^n}{\Delta t} = -u^n \cdot \nabla u^n + \nu \Delta u^n \quad \Rightarrow \tilde{u} \quad (4)$$

$$\Delta p^{n+1} = \frac{1}{\Delta t} \nabla \cdot \tilde{u} \quad \Rightarrow p^{n+1} \quad (5)$$

$$\frac{u^{n+1} - \tilde{u}}{\Delta t} = \nabla p^{n+1} \quad \Rightarrow u^{n+1} \quad (6)$$

上式中 u^n 表示第 n 时刻的速度值, \tilde{u}^n 表示速度的一个中间变量, p^n 表示第 n 时刻的压强值。

本方法的思想是把压强和速度分开解。使用的是将压强和速度分开来的时间分裂离散格式。从上式中我们可以看出第一步忽略压强项求解中间变量 \tilde{u} ; 第二步将第一步求解出来的 \tilde{u} 代入 Poisson 方程得到压强 p^{n+1} ; 第三步由前两步求解得到的中间变量 \tilde{u} 和压强 p^{n+1} 解出 u^{n+1} 。

对(6)变换如下:

$$\nabla \frac{u^{n+1} - \tilde{u}}{\Delta t} = \nabla \nabla p^{n+1} \tag{7}$$

上式与(5)相减得:

$$\nabla \cdot u^{n+1} = 0$$

所以整个解算过程是满足不可压缩条件的。

2.3 空间差分格式的实现

NaSt3DGP 基于 Chorin-type projection 方法, 采用有限差分格式对非均匀网格在空间离散化。对对流项的离散可以选用 Donor-Cell, HPLA, QUICK, SMART 和 VONOS 五种方法。在对低速流体的计算中, 比较困难的是对对流扩散项的处理和求解 Poisson 方程。

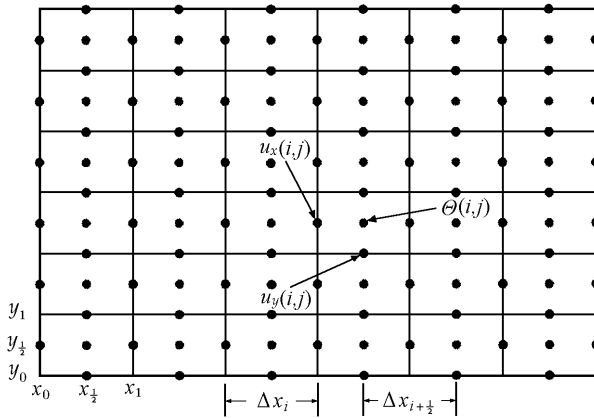


图 3 交错网格示意图

$u_x(i, j)$ 和 $u_y(i, j)$ 为速度分量, $\theta(i, j)$ 为温度, 三者所在的节点相互交错

Fig. 3 The staggered grid diagram

$u_x(i, j)$ and $u_y(i, j)$ are Velocity components, $\theta(i, j)$ is temperature, their grids are staggered

表 2 速度、温度、压力的坐标

Table 2 Velocity, temperature and pressure coordinates

各分量	坐标
$u_{i,j,k}$	$(x_i, y_{j+\frac{1}{2}}, z_{k+\frac{1}{2}})$
$v_{i,j,k}$	$(x_{i+\frac{1}{2}}, y_j, z_{k+\frac{1}{2}})$
$w_{i,j,k}$	$(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}}, z_k)$
$T_{i,j,k}$	$(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}}, z_k)$
$P_{i,j,k}$	$(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}}, z_{k+\frac{1}{2}})$

在介绍具体解法之前, 首先介绍一下本程序的网格系统。本程序采用交错网格的形式。如图 3 所示^[3], 速度在 x, y, z 三个方向的分量 u, v, w 、温度 T 及压力 P 分别定义在如表 2 所列的结点上。

空间差分格式: $\Delta x_i = x_i - x_{i-1}$, $\Delta x_{i+\frac{1}{2}} = \frac{x_i + x_{i+1}}{2}$; $\Delta y_j, \Delta y_{j+\frac{1}{2}}, \Delta z_k, \Delta z_{k+\frac{1}{2}}$ 分别在 y 方向和 z 方向也如此定义。

由于交错网格能成功地解决压力梯度离

散时所遇到的问题,该方法得到了广泛应用。现以 Poisson 方程的解法为例,来说明采用交错网格后,差分格式的特点。

在解 Poisson 方程时,为便于子区域数据传输且在子区域边界点实现差分格式,每一子区域均向相邻区域扩展一个网格,如图 4 所示^[3]。其他方向上的数据交换与此类似。

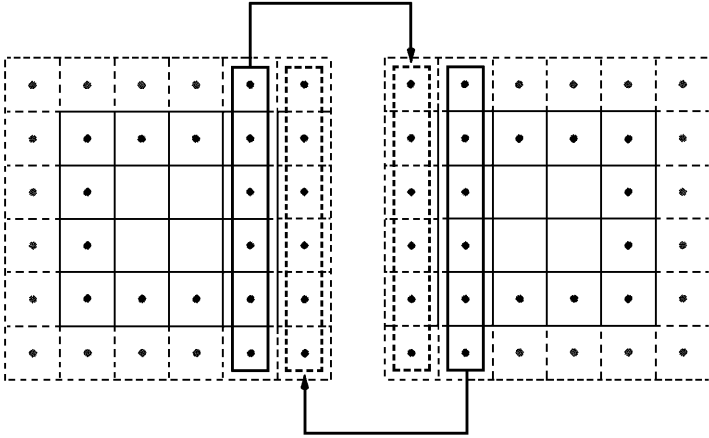


图 4 区域间数据传输示意图

Fig. 4 The diagram of regional data transmission

本程序中 Poisson 方程写成差分格式为:

$$\begin{aligned} \left[\frac{\partial^2 p}{\partial x^2} \right]_{i,j,k} &= \frac{1}{\Delta x_i} \left(\frac{p_{i+1,j,k} - p_{i,j,k}}{\Delta x_{i+\frac{1}{2}}} - \frac{p_{i,j,k} - p_{i-1,j,k}}{\Delta x_{i-\frac{1}{2}}} \right) \\ \left[\frac{\partial^2 p}{\partial y^2} \right]_{i,j,k} &= \frac{1}{\Delta y_j} \left(\frac{p_{i,j+1,k} - p_{i,j,k}}{\Delta y_{j+\frac{1}{2}}} - \frac{p_{i,j,k} - p_{i,j-1,k}}{\Delta y_{j-\frac{1}{2}}} \right) \\ \left[\frac{\partial^2 p}{\partial z^2} \right]_{i,j,k} &= \frac{1}{\Delta z_k} \left(\frac{p_{i,j,k+1} - p_{i,j,k}}{\Delta z_{k+\frac{1}{2}}} - \frac{p_{i,j,k} - p_{i,j,k-1}}{\Delta z_{k-\frac{1}{2}}} \right) \\ \left[\frac{\partial^2 p}{\partial x^2} \right]_{i,j,k} + \left[\frac{\partial^2 p}{\partial y^2} \right]_{i,j,k} + \left[\frac{\partial^2 p}{\partial z^2} \right]_{i,j,k} &= RHS[i][j][k] \end{aligned}$$

式中 $RHS[i][j][k]$ 是 $\nabla \cdot \tilde{u}/\Delta t$ 的离散形式。这是三维 7 点模式。程序中采用了 SOR、SSOR、Red-Black、8COLOR 及 BiCGSTA-B 等多种方法。其中 BiCGSTA-B 方法(预条件双共轭梯度法)精度最高收敛最快,是程序默认的解法。

2.4 软件包的构成及使用方法

NaSt3DGP 是一个标准的 UNIX 压缩包,在使用前需要输入如下命令进行解压缩: `gzip-de nast3dgp.tar.gz`——`tar xvf-` 或者 `tar xvzf nast3dgp.tar.gz`, 然后在新创建的子目录 `nast3dgp` 输入如下命令进行编译配置: `./configure`, 在编译完成后键入 `make` 或 `make install` 命令程序就可以运行了。如果想在 windows 操作系统下运行此程序,则只要安装一个软件 `cygwin` 就可以了,此软件模拟了 UNIX 的系统环境。

此软件包由以下几部分组成:源代码部分在 `src` 文件夹中,参数可在外部文件 `p.nav` 中修改,此程序还提供了 `matlab`、`VTK` 等图形接口。

3 在行星动力学中的应用

应用此软件对旋转星流体的非线性对流进行模拟,取得了令人满意的结果。

在研究行星流体对流问题时我们所选取的模型为:底部被均匀加热,且底部的温度要高于顶部温度的圆环。该圆环绕着竖直重力场方向以常角速度 Ω 匀速旋转,假设此圆环的宽度要远远小于圆环的外半径,则可忽略圆环的曲率,而将此圆环近似地看作在 x 方向具有周期性边界条件的方盒子(如图 5 所示)。图形中的 Γ 表示模型宽高之比; Ω 表示模型旋转的角速度。

需要求解的方程组为:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + T_{\alpha}^{-1} \boldsymbol{\kappa} \cdot \mathbf{u} = -\nabla p + R\Theta \boldsymbol{\kappa} + \nabla^2 \mathbf{u} \quad (8)$$

$$Pr \left(\frac{\partial \Theta}{\partial t} + \mathbf{u} \cdot \nabla \Theta \right) = \mathbf{u} \cdot \boldsymbol{\kappa} + \nabla^2 \Theta \quad (9)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (10)$$

上式中:

$$R = \frac{\alpha \beta g_0 d^4}{\kappa \mu}, \quad Pr = \frac{\nu}{\kappa}, \quad Ta = \tau^2 = \frac{4\Omega^2 d^4}{\mu^2}$$

在我们以后的讨论中,将使用参数 τ 代替泰勒数 Ta 作为旋转参数。参数中 β 是一个正常数。上下界面温差越高, β 越大,对流越容易形成。 $\boldsymbol{\kappa}$ 是平行于角速度 Ω 的垂直单位矢量。 α 为流体的膨胀系数。 κ 表示热扩散率。 ν 表示运动学粘性系数。 Θ 表示偏离纯传导状态的温度差。 R 是 Rayleigh 数,表示对流强度的物理量。当 R 小于某一个临界值 R_c 时,没有对流产生,此时流体只有热传导;当 R 超过临界值 R_c 时,就会有对流发生, R 越大,系统发生对流的强度就越大。 Pr 是 Prandtl 数,表示流体本身的一种性质,是动量扩散和热扩散的比值,是流体的特性。 τ 表示旋转系数。 τ 越大,系统旋转得越快。

3.1 程序修改后的检验

为了保证修改后程序的正确性,现分别用自检验和互检验的方法对其进行检验。

3.1.1 自检验

(1) 检验方法

自检验就是选取一个测试函数,如:

$$u_x = (\cos(x\pi) - 1) \sin(y\pi) \cos(2z\pi) \cos(t)$$

$$u_y = \sin(x\pi) (\cos(y\pi) - 1) \cos(2z\pi) \cos(t)$$

$$u_z = \sin(x\pi) \sin(y\pi) \sin(2z\pi) \cos(t)$$

$$\Theta = \sin(x\pi) \sin(y\pi) \cos(z\pi) \cos(t)$$

所选取的测试函数满足不可压缩假设,所以要求解的控制方程减少为两个:

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} + \tau \boldsymbol{\kappa} \cdot \mathbf{u} = -\nabla p + R \boldsymbol{\kappa} \Theta + \nabla^2 \mathbf{u} + f \quad (11)$$

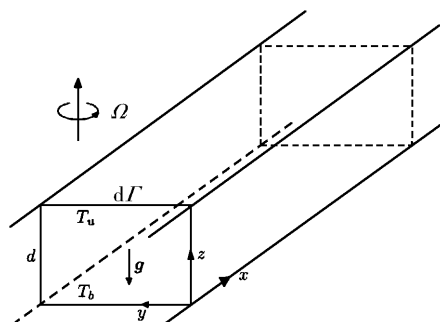


图 5 简化模型示意图

Fig. 5 Schematic Model Simplification

$$\text{Pr} \left(\frac{\partial \Theta}{\partial t} + (\mathbf{u} \cdot \nabla) \Theta \right) = \kappa \cdot \mathbf{u} + \nabla^2 \Theta + \theta \quad (12)$$

将选取的测试函数代入上述方程中后,可以得到 f 和 θ 的表达式。将这两个表达式代回控制方程(11)和(12)消去 f 和 θ 可得一组新控制方程。这时,我们选取的测试函数一定是新方程组的一组解析解。将 NaSt3DGP 程序解算新方程组得到的温度场、速度场与测试函数进行对比,即可完成自洽检验。

在进行自洽检验时,我们选取的边界条件和初值(此初值要满足不可压缩假设)分别如下:

边界条件:选取模型长度为 4,模型的宽度为 4,高度为 1。

x 方向:周期边界条件;

y 方向: $\mathbf{u} = 0$; $\frac{\partial \Theta}{\partial y} = 0$;

z 方向: $\frac{\partial u_z}{\partial z} = \frac{\partial u_y}{\partial z} = u_z$, $\Theta = 0$

初值:

$$u_x = (\cos(x\pi) - 1) \sin(y\pi) \cos(2z\pi)$$

$$u_y = \sin(x\pi) (\cos(y\pi) - 1) \cos(2z\pi)$$

$$u_z = \sin(x\pi) \sin(y\pi) \sin(2z\pi)$$

$$\Theta = \sin(x\pi) \sin(y\pi) \cos(z\pi)$$

(2) 检验结果

现对测试函数给出的解析解和通过本程序求解的数值解进行比较。为了更直观地说明问题,本文给出两种解在 $z = 0.5$ 平面的 u_x 、 Θ 等值线图。从图 6 和图 7 中可以看到由本程序求解出来的速度场和温度场与测试函数给出的速度场和温度场的值是吻合的(限于篇幅本文没有给出速度另外两个分量的数值解与解析解的比较图,但它们也是一致的)。这说明修改后的程序是自洽的。

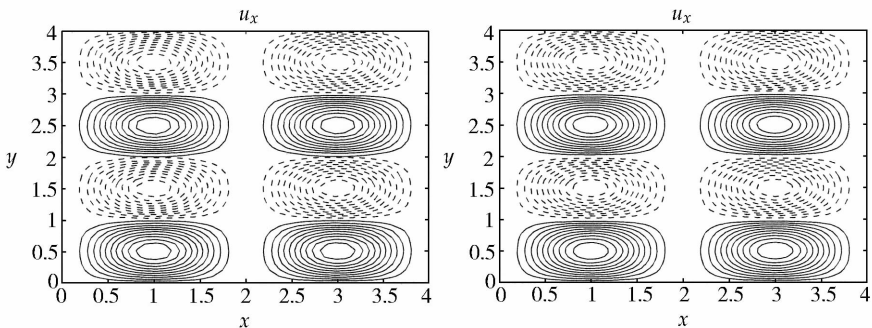


图 6 自洽检验时 u_x 的解析解 (左) 和数值解(右)等值线图

Fig. 6 Contours of analytical solutions of u_x (left panel) and contours of numerical solution u_x (right panel) in itself test

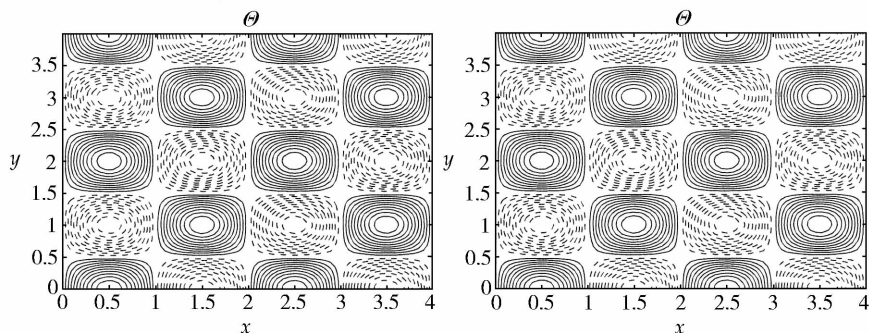


图 7 自洽检验时 Θ 的解析解 (左) 和数值解(右)等值线图

Fig. 7 Contours of analytical solutions of Θ (left panel) and contours of numerical solution Θ (right panel) in itself test

3.1.2 互检验

(1) 检验方法

程序的互检验就是将用修改过的程序计算得到的速度场、温度场的数值解和通过线性理论得到的速度场、温度场的解析解进行比对。前者是非线性数值解,后者是线性解析解,二者之所以能够进行比对是因为算例中 R 与 R_c 很接近 (R 是数值模拟计算时所使用的 Rayleigh 数, R_c 是由线性理论得到的发生对流时的临界 Rayleigh 数),所以在数值模拟时的非线性效应不强,可以将其结果与线性理论得到的结果相比较。

(2) 检验结果

用上一节所给出的边界条件计算了 2 组算例。把两者的 $z = 0.5$ 平面上 u_z, Θ 的等值线图进行比较(分别见图 8,9 和图 10,11)。具体算例分述如下:

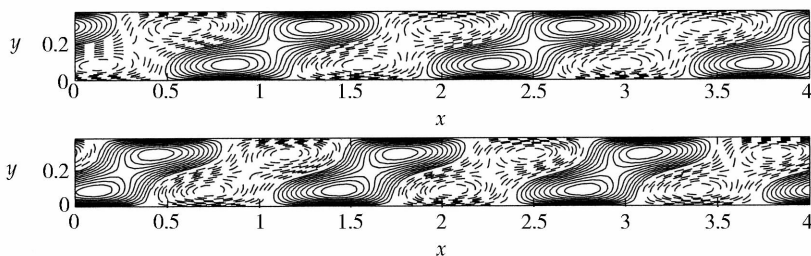


图 8 互检验算例(1)中 u_z 解析解 (上) 及其数值解(下)等值线图

Fig. 8 Contours of analytical solutions of u_z (up) and contours of numerical solution u_z (bottom) in mutual test one

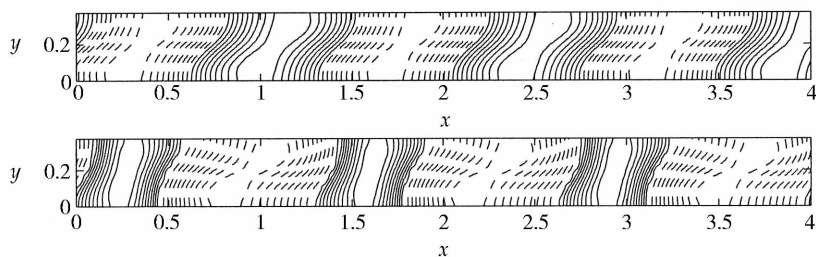


图 9 互检验算例(1)中 Θ 解析解(上) 及其数值解(下)等值线图

Fig. 9 Contours of analytical solutions of Θ (up) and contours of numerical solution Θ (bottom) in mutual test one

a) $\Gamma = 0.375$, $\tau = 10^3$, $\text{Pr} = 7$, $R_c = 2.44 \times 10^4$, $R = 2.6 \times 10^4$ 。

从图 8 和图 9 中可以看出在狭窄模型(当 $\Gamma \leq O(\tau^{-1/3})$) 时)中解析解和数值解吻合得很好(两个图形在 x 轴方向有位移,这是因为我们在 x 方向选取的边界条件是周期性边界条件,所以实质上二者是完全相同的)。

b) $\Gamma = 2$, $\tau = 10^3$, $\text{Pr} = 7$, $R_c = 3.53 \times 10^4$, $R = 3.7 \times 10^4$ 。

从图 10 到图 11 中可以看出在宽模型(当 $\Gamma \geq O(\tau^{-1/3})$) 时)中解析解和数值解的图形吻合得也很好。需要说明的是在宽模型中发生边界流的时候有两组解共同存在,这两组解恰好

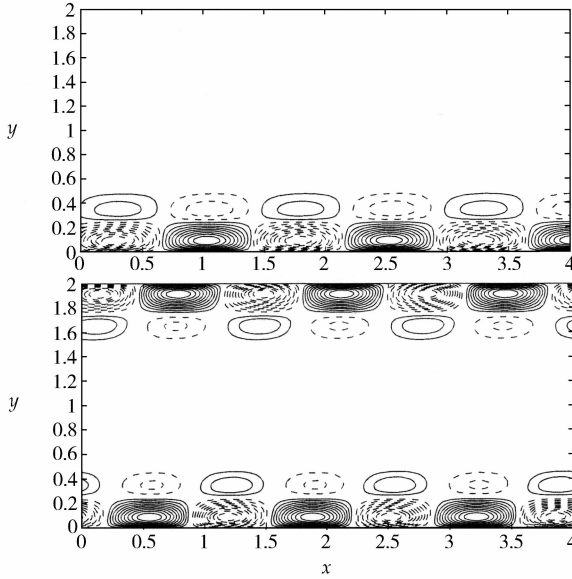


图 10 互检验算例(2)中 u_z 解析解(上)及其数值解(下)等值线图

Fig. 10 Contours of analytical solutions of u_z (up) and contours of numerical solution u_z (bottom) in mutual test two

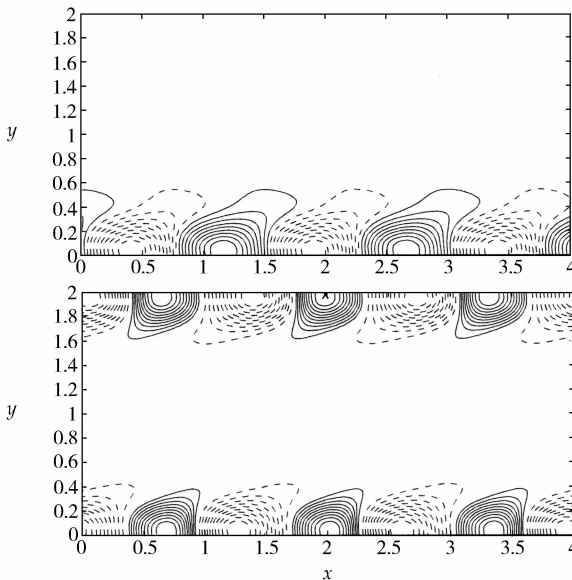


图 11 互检验算例(2)中 θ 解析解(上)及其数值解(下)等值线图形

Fig. 11 Contours of analytical solutions of θ (up) and contours of numerical solution θ (bottom) in mutual test two

是方向相反的。我们给出的解析解的等值线图形仅仅是其中的一组解。

通过对修改后程序进行的自检及互检,可以证明修改后程序的可靠性。

3.2 旋转行星流体非线性动力学特性数值模拟

用解析的方法很难得到旋转行星内部的非线性对流,特别是强非线性对流。现用修改后的程序对旋转行星的非线性对流问题进行研究。

本节对旋转模型中的非线性对流进行了数值模拟。该模型的上下表面均为 stress-free 边界。以后凡未指明何处的边界条件均指模型的侧边界。这些数值模拟集中于如下两种不同的情况:宽模型($\Gamma \geq O(\tau^{-1/3})$)中,1) stress-free 边界条件下发生的边界流;2) non-slip 边界条件下发生的定常对流。

就这两种情况,本文模拟了不同 Prandtl 数及不同 Rayleigh 数下发生的非线性对流。模拟结果显示了从定常对流到湍流过程中的各种非线性效应。

本节对宽模型 stress-free 边界条件下的 12 组算例进行了数值模拟。具体算例信息参见表 3。表中 R_c 表示发生对流时的临界 Rayleigh 数,这是通过线性理论计算出来的。

表 3 宽模型 stress-free 边界条件下算例表

Table 3 A summary of the runs for numerical simulations in a wide channel with the stress-free sidewalls

算例号	Γ	Pr	R_c	R
1	2.0	7.0	2.55×10^4	4.0×10^4
2	2.0	7.0	2.55×10^4	7.0×10^4
3	2.0	7.0	2.55×10^4	1.0×10^5
4	2.0	7.0	2.55×10^4	2.5×10^5
5	2.0	7.0	2.55×10^4	5.0×10^5
6	2.0	7.0	2.55×10^4	1.0×10^6
7	2.0	7.0	2.55×10^4	2.5×10^6
8	2.0	0.7	2.55×10^4	4.0×10^4
9	2.0	0.7	2.55×10^4	1.0×10^5
10	2.0	0.7	2.55×10^4	5.0×10^5
11	2.0	0.7	2.55×10^4	1.0×10^6
12	2.0	0.7	2.55×10^4	2.5×10^6

模拟结果参见图 12 ~ 15。

从图形 12 ~ 15 可以得出如下结论:当 R 比较小的时候,无论 Prandtl 数为 7.0 还是 0.7,均只发生边界流(见图 12(左)和图 14(左));但当 R 增大到大约 $10 R_c$ 时,边界流和内部对流同时发生(见图 12(右)和图 14(右));当 R 增大到大约 $100 R_c$ 时就会发生湍流(见图 13 和图 15)。

把此程序应用到宽模型($\Gamma \geq O(\tau^{-1/3})$)中,non-slip 边界条件下,通过模拟结果(由于篇幅原因在这里不能把结果一一列出)可发现速度边界条件在决定强非线性对流主要性质时并没有起很重要的作用。此结论在地球物理学和天体物理学的应用方面是非常有意义的^[1]。

根据需要应用此程序可以模拟在不同的宽高之比(即不同 Γ 值)的模型中流体发生对流的情况。

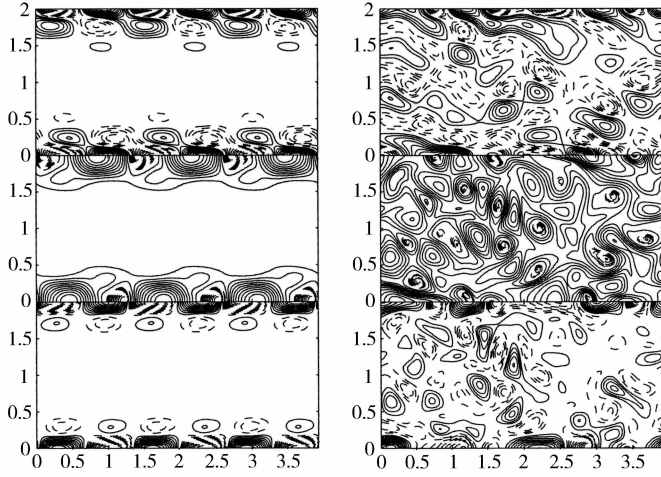


图 12 xy 平面上 u_x (顶部)、 Θ (中部)、 u_z (下部) 的等值线图

$R = 7 \times 10^4$ (左), $R = 2.5 \times 10^5$ (右), $\tau = 10^3$, $\Gamma = 2$, $\text{Pr} = 7.0$, stress-free 边界条件

Fig. 12 In each panel, contours of u_x (top), Θ (middle) and u_z (bottom) are shown in a horizontal xy plane for $\tau = 10^3$, $\Gamma = 2$, $\text{Pr} = 7.0$. The Rayleigh numbers are $R = 7 \times 10^4$ (left panel), $R = 2.5 \times 10^5$ (right panel). The sidewalls are stress-free

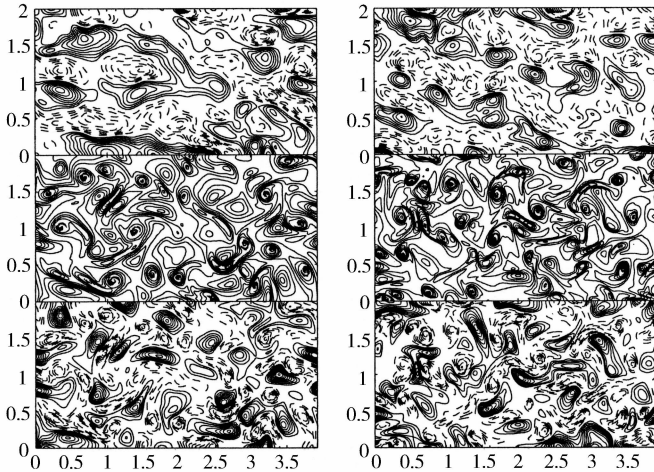


图 13 xy 平面上 u_x (顶部)、 Θ (中部)、 u_z (下部) 的等值线图

$R = 1 \times 10^6$ (左), $R = 2.5 \times 10^6$ (右), $\tau = 10^3$, $\Gamma = 2$, $\text{Pr} = 7.0$, stress-free 边界条件

Fig. 13 In each panel, contours of u_x (top), Θ (middle) and u_z (bottom) are shown in a horizontal xy plane for $\tau = 10^3$, $\Gamma = 2$, $\text{Pr} = 7.0$. The Rayleigh numbers are $R = 1 \times 10^6$ (left panel), $R = 2.5 \times 10^6$ (right panel). The sidewalls are stress-free

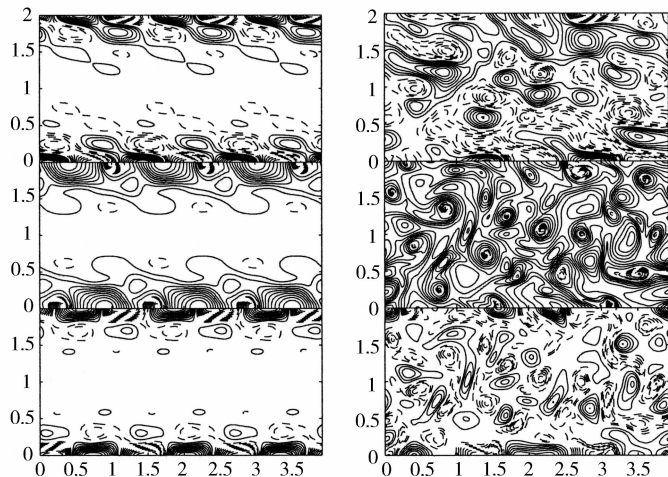


图 14 xy 平面上 u_x (顶部)、 Θ (中部)、 u_z (下部) 的等值线图

$R = 4 \times 10^4$ (左), $R = 1 \times 10^5$ (右), $\tau = 10^3$, $\Gamma = 2$, $Pr = 0.7$, stress-free 边界条件

Fig. 14 In each panel, contours of u_x (top), Θ (middle) and u_z (bottom) are shown in a horizontal xy plane for $\tau = 10^3$, $\Gamma = 2$, $Pr = 0.7$. The Rayleigh numbers are $R = 4 \times 10^4$ (left panel), $R = 1 \times 10^5$ (right panel). The sidewalls are stress-free

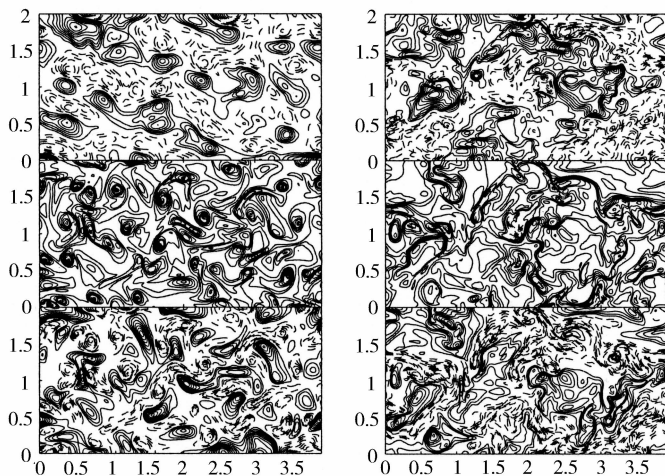


图 15 xy 平面上 u_x (顶部)、 Θ (中部)、 u_z (下部) 的等值线图

$R = 1 \times 10^6$ (左), $R = 2.5 \times 10^6$ (右), $\tau = 10^3$, $\Gamma = 2$, $Pr = 0.7$, stress-free 边界条件

Fig. 15 In each panel, contours of u_x (top), Θ (middle) and u_z (bottom) are shown in a horizontal xy plane for $\tau = 10^3$, $\Gamma = 2$, $Pr = 0.7$. The Rayleigh numbers are $R = 1 \times 10^6$ (left panel), $R = 2.5 \times 10^6$ (right panel). The sidewalls are stress-free

4 程序的优点及局限性

本程序的主要优点有：

- (1) 由于可以进行并行计算, 所以对于网格点数较大的计算可以提高运算速度。
- (2) 采用区域分解的算法, 并行效率较高。
- (3) 可以求解非均匀网格问题。
- (4) 功能强大, 可以解很复杂的流体力学问题。
- (5) 提供了环境变量文件, 这样更方便修改参数及一些环境变量。
- (6) 开源软件。可以根据自己的需要进行修改, 自由度高。
- (7) 为多种画图软件提供了数据接口。

本程序中存在的局限性是：

- (1) 由于差分格式是二阶显式的方法, 所以很难达到更高精度。
- (2) 差分方法虽然比较简单, 但稳定性和计算效率比有限元等办法要差, 特别是在非线性情况下更是如此。
- (3) 因该软件运行中存在大量数据通讯, 故对并行计算机的网络速度要求较高。

参 考 文 献

- [1] Chang Yingli, Liao Xinhao, Zhang Keke. Geophysical and Astrophysical Fluid Dynamics, 2006, 215 ~ 241
- [2] 吕涛等. 区域分解算法 - 偏微分方程数值新技术, 北京: 科学出版社, 1992
- [3] Griebel M, Croce R, Engel M. Nast3dgp a parallel 3d navier-stokes solver user's guide. 1977

INTRODUCTION AND APPLICATION OF THE PARALLEL 3D FLUID DYNAMICS SOFTWARE PACKAGE NaSt3DGP

CHANG Ying-li

(Shanghai Fisheries University, Shanghai 200090, China)

Abstract

NaSt3DGP is a parallel software package to solve the 3D incompressible fluid dynamic problems in Cartesian coordinates by using Finite Difference Method. This paper described the software's principles of domain-decomposition policy, finite difference scheme and flow chart in detail. Its reliability and precision are tested by our applications and the efficiency is appraised.

Key words decomposition — difference stencils — program test