不可压缩流动的并行数值方法

尚月强^①, 何银年^{②*}

① 贵州师范大学数学与计算机科学学院, 贵阳 550001;

② 西安交通大学数学与统计学院, 西安 710049

E-mail: yueqiangshang@gmail.com, heyn@mail.xjtu.edu.cn

收稿日期: 2012-01-09; 接受日期: 2012-11-07; * 通信作者

国家自然科学基金 (批准号: 11001061 和 11271298) 和贵州省科学技术基金 (批准号: 20132212) 资助项目

摘要 不可压缩流动的数值模拟是计算流体力学的重要组成部分. 基于有限元离散方法, 本文设计了不可压缩 Navier-Stokes (N-S) 方程支配流的若干并行数值算法. 这些并行算法可归为两大类: 一类是基于两重网格离散方法, 首先在粗网格上求解非线性的 N-S 方程, 然后在细网格的子区域上并行求解线性化的残差方程, 以校正粗网格的解; 另一类是基于新型完全重叠型区域分解技巧, 每台处理器用一局部加密的全局多尺度网格计算所负责子区域的局部有限元解. 这些并行算法实现简单, 通信需求少, 具有良好的并行性能, 能获得与标准有限元方法相同收敛阶的有限元解. 理论分析和数值试验验证了并行算法的高效性.

关键词 不可压缩流动 Navier-Stokes 方程 有限元 并行算法 区域分解 两重网格

MSC (2010) 主题分类 76D15, 35Q14

1 引言

不可压缩流动是常见的流动现象, 其数值模拟计算是计算流体力学的重要组成部分. 因此, 不可压缩流动数值模拟方法的研究一直是计算流体力学领域的热门课题之一 (参见文献 [1-5]). 非定常不可压缩 Newton 黏性流的控制方程为如下的 Navier-Stokes (N-S) 方程:

$$\mathbf{u}_t - \nu \Delta \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} + \nabla p = \mathbf{f}, \quad \text{在 } \Omega \times [0, T] \text{ 内}, \quad (1.1)$$

$$\operatorname{div} \mathbf{u} = 0, \quad \text{在 } \Omega \times [0, T] \text{ 内}, \quad (1.2)$$

$$\mathbf{u} = \mathbf{0}, \quad \text{在 } \partial\Omega \times [0, T] \text{ 上}, \quad (1.3)$$

$$\mathbf{u}|_{t=0} = \mathbf{u}_0, \quad \text{在 } \Omega \text{ 内}, \quad (1.4)$$

其中 $\Omega \subset \mathbb{R}^n$ ($n = 2, 3$) 是流体流动的区域, $\mathbf{u} = (u_1, \dots, u_n)^T$ 是流体速度, p 是压力, $\mathbf{f} = (f_1, \dots, f_n)^T$ 是事先给定的作用力, ν 是黏性系数, T 是有限时间. 若上述方程与时间无关 (即无时间导数项 $\mathbf{u}_t$), 则为定常 N-S 方程.

不可压缩 N-S 方程是典型的非线性方程组, 其数值模拟面临两大困难: 所需存储空间大和计算时间长. 因此, 需借助并行计算技术, 以实现 N-S 方程的快速求解, 进而实现不可压缩流动的大规模数值模拟. 然而, 要实现不可压缩流动的并行数值模拟, 除了高性能并行机和一套完善的并行编程系统软

件外, 高效的并行算法更是不可缺少的重要组成部分, 它对提高并行机的运行效率、确保数值模拟结果的有效性和准确性起着至关重要的作用。

基于以上认识, 国内外一大批研究者和工程技术人员投身到 N-S 方程的并行数值算法研究, 并取得了一系列研究成果. 已有的并行数值方法大致可分为两大类: 并行离散方法^[6-11] 和离散所得代数方程组的并行求解方法^[12-16]. 并行策略包括基于单元级的 EBE (Element-by-element)^[17], EDE (Edge-by-edge)^[18], NBN (Node-by-node)^[19] 方法、重叠型区域分解^[20-23] 与非重叠型区域分解算法^[24-27] 和并行两重或多重网格方法^[28-33] 以及并行自适应方法^[34-36] 等。

对于一个并行数值算法而言, 除了应具有通常意义下一个数值算法应具有数值特征 (稳定性、收敛性和准确性等) 和良好的并行性外, 较少的通信需求和较容易的编程实现更是应考虑的重要因素. 一方面, 对于实际应用而言, 一个复杂的并行算法所需的程序代码的编写、调试时间和处理器之间的通信时间可能大大超过并行计算所减少的计算时间; 另一方面, 由于并行计算软件的开发远远落后于硬件的发展, 使得并行编程远比传统的串行编程困难, 特别是对于复杂流动的数值模拟. 因此, 设计出实现简单、通信需求少的求解 N-S 方程的并行数值算法对不可压缩流动的数值模拟显得尤为重要. 基于此认识, 我们使用有限元离散方法、基于重叠型区域分解技巧和两重网格离散方法, 提出并研究了 N-S 方程简单而高效的并行离散方法. 这些方法实现简单, 通信需求少, 具有良好的并行性能。

2 基于 Oseen 线性化的有限元并行算法

我们将求解 N-S 方程的 Picard 迭代法和区域分解算法相结合, 提出了求解定常 N-S 方程的并行 Oseen 线性化的有限元方法. 该方法主要是基于对 N-S 方程有限元解全局和局部性质的认识, 即其低频分量可用粗网格较好地逼近, 而高频分量可在细网格上局部和并行计算. 我们首先在一粗网格上用 Oseen 迭代法求解非线性的 N-S 方程 (每一迭代步求解 Oseen 方程), 然后在细网格的相互重叠的子区域上并行求解一线性化的 Oseen 方程以校正粗网格的解. 使用有限元局部误差分析的理论工具, 我们对所得有限元解的误差界进行了分析, 得到如下理论结果^[37]:

$$\begin{aligned} \|\nabla(\mathbf{u} - \mathbf{u}_m^h)\|_{0,\Omega} + \|p - p_m^h\|_{0,\Omega} \leq c \left(h^s + H^{s+1} \left(1 + \frac{\|\mathbf{f}\|_{0,\Omega}}{\|\mathbf{f}\|_{-1,\Omega}} \right) \right) \|\mathbf{f}\|_{s-1,\Omega} \\ + C \left(\frac{N}{\nu^2} \|\mathbf{f}\|_{-1,\Omega} \right)^m \|\mathbf{f}\|_{-1,\Omega}, \quad 1 \leq s \leq k, \quad m \geq 0, \end{aligned} \quad (2.1)$$

$$\begin{aligned} \|\nabla(\mathbf{u} - \mathbf{u}_m^h)\|_{0,\Omega} + \|p - p_m^h\|_{0,\Omega} \leq c \left(h^s + H^{s+1} \left(1 + \frac{\|\mathbf{f}\|_{0,\Omega}}{\|\mathbf{f}\|_{-1,\Omega}} \right) \right) \|\mathbf{f}\|_{s-1,\Omega} \\ + C \|\mathbf{u}_H^m - \mathbf{u}_H^{m-1}\|_{0,\Omega}, \quad 1 \leq s \leq k, \quad m \geq 1, \end{aligned} \quad (2.2)$$

其中 $\|\cdot\|_{0,\Omega}$ 是分片范数, $(\mathbf{u}, p)$ 是 N-S 方程的准确解, $(\mathbf{u}_m^h, p_m^h)$ 是并行算法所得近似解, m 是粗网格上非线性迭代次数, $\mathbf{u}_H^m$ 是粗网格上非线性方程的第 m 次迭代解, H 和 h 分别是粗细网格的尺寸, $N > 0$ 是只依赖于求解区域的常数, s 是分片插值多项式的次数.

估计式 (2.1) 和 (2.2) 表明该并行算法所得有限元解的误差主要来源于两部分, 一部分是粗细网格上的离散误差, 另一部分是非线性迭代误差. 该结果表明, 若选取粗细网格的尺寸使其满足 $H = \mathcal{O}(h^{\frac{s}{s+1}})$, 则我们的方法可获得与标准有限元方法相同的收敛阶. 但一方面, 我们的算法在细网格上只需求解一线性问题; 另一方面, 细网格上的线性问题是并行计算的. 所以, 与标准有限元方法相比,

我们的并行算法能大大节约计算时间. 针对二维区域 $\Omega = [0, 1] \times [0, 1]$, N-S 方程准确解为

$$\begin{aligned} u_1 &= 10x^2(x-1)^2y(y-1)(2y-1), \\ u_2 &= -10y^2(y-1)^2x(x-1)(2x-1), \\ p &= 10(2x-1)(2y-1) \end{aligned} \quad (2.3)$$

的数值算例, 使用 Taylor-Hood 有限元, 将求解区域分解成 4 个子区域

$$\begin{aligned} D_1 &= (0, 1/2) \times (0, 1/2), \quad D_2 = (1/2, 1) \times (0, 1/2), \\ D_3 &= (0, 1/2) \times (1/2, 1), \quad D_4 = (1/2, 1) \times (1/2, 1) \end{aligned}$$

的数值结果验证了理论分析的正确性 (见表 1, 其中 m 为粗网格上非线性迭代次数).

估计式 (2.2) 右端项第二项不但指明了非线性迭代对有限元解精度的影响, 还给出了一个迭代收敛准则:

$$\|\mathbf{u}_H^n - \mathbf{u}_H^{n-1}\|_{0,\Omega} \leq ch^s. \quad (2.4)$$

这是一个十分有用的结果. 我们使用不同的迭代收敛准则进行了数值试验, 结果表明该迭代收敛准则是有效的和可靠的, 如图 1 所示.

我们还将该算法与其他两个相似算法 (基于 Newton 校正^[38] 和 Stokes 校正^[39] 的并行算法) 进行了数值比较, 结果表明, 若同时考虑计算时间和所得有限元解的精度, 我们新设计的并行算法优于其他两种算法, 见表 2.

表 1 基于 Oseen 线性化的有限元并行算法数值结果^[37]

h	H	m	$\frac{\ \nabla(\mathbf{u} - \mathbf{u}_m^h)\ _{0,\Omega}}{\ \nabla \mathbf{u}\ _{0,\Omega}}$	$\frac{\ p - p_m^h\ _{0,\Omega}}{\ p\ _{0,\Omega}}$	收敛率
1/8	1/4	2	0.049645900	0.016829400	—
1/27	1/9	2	0.005326320	0.001422360	1.98773
1/64	1/16	2	0.001265250	0.000221532	2.01543
1/125	1/25	2	0.000401114	6.63533e-05	1.77248
1/216	1/36	2	0.000157156	2.31710e-05	1.84895

图 1 基于 Oseen 线性化的有限元并行算法速度的 H^1 -误差随非线性迭代的变化情况

表 2 基于 Oseen 线性化的有限元并行算法数值结果比较 [40]

ν	非线性迭代次数	Newton 校正算法		Oseen 校正算法		Stokes 校正算法	
		CPU(s)	$\frac{ \nabla(\mathbf{u}-\mathbf{u}^h) _{0,\Omega}}{\ \nabla\mathbf{u}\ _{0,\Omega}}$	CPU(s)	$\frac{ \nabla(\mathbf{u}-\mathbf{u}^h) _{0,\Omega}}{\ \nabla\mathbf{u}\ _{0,\Omega}}$	CPU(s)	$\frac{ \nabla(\mathbf{u}-\mathbf{u}^h) _{0,\Omega}}{\ \nabla\mathbf{u}\ _{0,\Omega}}$
1.0000	2	4.37	0.00126526	3.97	0.00126525	3.97	0.00126526
0.1000	3	4.74	0.00126534	4.57	0.00126533	4.30	0.00126543
0.0100	4	5.06	0.00126619	4.92	0.00126635	4.62	0.00127342
0.0010	8	6.39	0.00127988	6.28	0.00129384	5.95	0.00187660
0.0005	10	7.08	0.00129826	7.01	0.00133754	6.62	0.00322106
0.0001	17	9.38	0.00163602	9.23	0.00209510	8.96	0.02749690

若让所有处理器同时计算粗网格上的有限元解, 则在整个计算过程中, 除了最后计算结果的组装和可视化外, 处理器之间无需通信, 因而该算法通信需求少, 具有良好的并行性能. 曙光并行机上的数值试验取得了超线性的加速比和并行效率, 如图 2 所示, 其中加速比与并行效率按以下公式计算:

$$S_p = \frac{T(2)}{T(J)}, \quad E_p = \frac{2 \times T(2)}{J \times T(J)}, \quad (2.5)$$

此处, $T(2)$ 和 $T(J)$ 分别为处理器个数为 2 和 J 时并程序运行时间 (墙上时间). 图 3 则给出了 8 个和 12 个子区域时并行算法模拟二维方腔驱动流的数值结果.

使用 Newton 迭代法求解粗网格上的非线性方程, 然后在细网格的子区域上并行求解 Oseen 线性化的残差方程以校正粗网格的解, 我们得到另一并行算法. 并通过理论分析, 得到如下误差估计 [41]:

$$\begin{aligned} |||\nabla(\mathbf{u} - \mathbf{u}_m^h)|||_{0,\Omega} + |||p - p_m^h|||_{0,\Omega} \leq & c \left(h^s + H^{s+1} \left(1 + \frac{\|\mathbf{f}\|_{0,\Omega}}{\|\mathbf{f}\|_{-1,\Omega}} \right) \right) \|\mathbf{f}\|_{s-1,\Omega} \\ & + C |\log H|^{1/2} \|\nabla(\mathbf{u}_H^m - \mathbf{u}_H^{m-1})\|_{0,\Omega} \|\mathbf{u}_H^m - \mathbf{u}_H^{m-1}\|_{0,\Omega}, \end{aligned} \quad (2.6)$$

其中 $1 \leq s \leq k$, $m \geq 1$. (2.6) 与 (2.2) 一样, 也给出了一个粗网格上非线性迭代的收敛准则:

$$|\log H|^{1/2} \|\nabla(\mathbf{u}_H^n - \mathbf{u}_H^{n-1})\|_{0,\Omega} \|\mathbf{u}_H^n - \mathbf{u}_H^{n-1}\|_{0,\Omega} \leq ch^s. \quad (2.7)$$

数值实验验证了该收敛准则的有效性和可靠性, 如图 4 所示.

图 2 基于 Oseen 线性化的有限元并行算法的加速比 (a) 和并行效率 (b)

图 3 基于 Oseen 线性化的有限元并行算法二维方腔驱动流时 8 个子区域 (a) 和 12 个子区域 (b) 计算所得的速度场流线

图 4 基于 Newton 迭代的 Oseen 线性化有限元并行算法速度的 H^1 -误差 ((a), (b)) 和压力的 L^2 -误差 ((c), (d)) 随非线性迭代的变化情况^[41]

针对三维方腔驱动流问题, 我们将求解区域 $\Omega = [0, 1] \times [0, 1] \times [0, 1]$ 分解成大小相同的 8 个子区域 (如图 5 所示), 对算法进行了数值实验. 图 6 描述了 Reynolds 数 $Re = 100$ 时计算结果在 $y = 0.2, 0.4, 0.6$ 和 0.8 处的平面切图, 从中可以看出算法的有效性.

图 5 三维方腔驱动流的区域分解

图 6 基于 Newton 迭代的 Oseen 线性化有限元并行算法三维方腔驱动流 $Re = 100$ 时速度场在 $y = 0.2, 0.4, 0.6$ 和 0.8 处的平面切图

3 基于压力 Dirichlet 边界条件的并行两重网格离散方法

使用两重网格离散方法, 我们提出了一种新的求解定常 N-S 方程的有限元并行算法. 该算法主要讨论细网格的子区域上, 局部 N-S 方程的速度和压力的边界条件的选取以及如何校对粗网格解, 从而得到较高精度的有限元解.

众所周知, N-S 方程的压力不是唯一确定的, 相差一个常数. 为了确保压力的唯一性, 通常的做法是取压力属于 L_0^2 空间或固定压力在某一点的值. 针对已有相关算法在粗细网格上的压力都限制在 L_0^2 空间, 但由于子区域的相互重叠, 使得某些问题的数值模拟结果不大理想的情况, 我们提出了一种新的有限元并行算法. 该算法首先在粗网格上求解非线性的 N-S 方程, 然后在相互重叠的细网格的子区域上并行求解基于 Newton 迭代法线性化的残差方程 (即计算一个 Newton 迭代步), 以校正粗网格的有限元解. 该算法区别于已有的相关算法在于在粗网格上限制压力属于 L_0^2 空间, 而在细网格相互重叠的子区域上求解粗网格解的残差方程时, 在人工边界上给压力一齐次 Dirichlet 边界条件 (即 $p = 0$), 进而取得较现有算法好的性能^[42]. 我们对已知解析解的三维问题、二维方腔驱动流和圆柱扰流进行了数值模拟, 验证了算法的有效性.

针对

$$\Omega = [0, 1] \times [0, 1] \times [0, 1],$$

准确解为

$$\begin{aligned} u_1 &= x^2(x-1)^2[2y(y-1)(2y-1)z^2(z-1)^2 - 2y^2(y-1)^2z(z-1)(2z-1)], \\ u_2 &= y^2(y-1)^2[-2x(x-1)(2x-1)z^2(z-1)^2 + 2x^2(x-1)^2z(z-1)(2z-1)], \\ u_3 &= z^2(z-1)^2[2x(x-1)(2x-1)y^2(y-1)^2 - 2x^2(x-1)^2y(y-1)(2y-1)], \\ p &= x^2 + y^2 + z^2 - 1 \end{aligned} \quad (3.1)$$

的三维 N-S 方程 (其中 $\nu = 0.1$), 将区域分解成大小相同的 8 个子区域 (如图 5 所示), 使用 Taylor-Hood 元, 我们对算法进行了数值实验, 数值结果表明算法取得了渐近最优收敛阶 (见表 3, 其中 m 为粗网格上的非线性迭代次数, (u_m^h, p_m^h) 为并行算法所得有限元解).

我们将所设计的并行算法与亏量 - 校正方法相结合 (即使用亏量 - 校正方法求解粗网格上的非线性方程), 对二维方腔驱动流问题进行了数值试验. 图 7-9 分别描述了当 $Re = 5000, 7500, 10000$ 时, 我们的并行算法 16 个子区域时计算所得的、经过区域中心的水平线和垂直线上的速度与文献 [43] 中数据的比较, 从中可以看出我们的并行算法的有效性.

就圆柱扰流问题, 我们将求解区域分成 6 个子区域, 使用 Taylor-Hood 元进行计算, 所得结果如图 10 所示 (圆柱附近的局部图), 该数值结果验证了算法的有效性.

表 3 基于压力 Dirichlet 边界条件的并行两重网格离散方法数值结果

h	H	m	$\ \nabla(\mathbf{u} - \mathbf{u}_m^h)\ _{0,\Omega}$	$\ p - p_m^h\ _{0,\Omega}$	收敛率
1/4	1/2	2	0.08616100	0.02673870	—
1/8	1/4	2	0.02088550	0.00742336	2.00
1/12	1/6	2	0.00825844	0.00343394	2.18
1/16	1/8	2	0.00419274	0.00196492	2.23

图 7 基于压力 Dirichlet 边界条件的并行两重网格离散方法方腔驱动流 $Re = 5000$ 时的数值模拟结果比较: 速度 u_1 (a), 速度 u_2 (b)

图 8 基于压力 Dirichlet 边界条件的并行两重网格离散方法方腔驱动流 $Re = 7500$ 时的数值模拟结果比较: 速度 u_1 (a), 速度 u_2 (b)

图 9 基于压力 Dirichlet 边界条件的并行两重网格离散方法方腔驱动流 $Re = 10000$ 时的数值模拟结果比较: 速度 u_1 (a), 速度 u_2 (b)

图 10 基于压力 Dirichlet 边界条件的并行两重网格离散方法圆柱扰流 $Re = 20$ ((a), (b)) 和 $Re = 40$ ((c), (d)) 时数值模拟结果: 速度场流线 ((a), (c)) 与压力 ((b), (d))

4 基于完全重叠型区域分解的有限元并行算法

基于一种新型完全重叠型区域分解技巧, 我们研究了求解 N-S 方程的并行有限元方法. 该方法中, 我们首先将求解区域分解成若干个子区域, 让每台处理器负责一个区域, 各自对所负责的子区域生成尺度为 h 的细网格, 并对其余区域生成尺度为 $H \gg h$ 的粗网格, 然后采用自适应处理等技巧使得粗细网格在交界处相容 (即任两个单元的交集或者为空集, 或者为一公共顶点, 或者为一公共边或为一公共面). 图 11 给出了二维、两个子区域情形的完全重叠型区域分解及相应的网格剖分情况. 各个处理器用这样的全局多尺度网格求解相应的子问题, 计算结束时只需取互不重叠的子区域部分的结果, 而舍去子区域以外的计算结果. 由于每个子问题是一个全局问题, 它定义在整个求解区域上, 但绝大部分自由度来自其所负责的子区域, 使得算法实现简单, 通信需求少^[44, 45].

我们分别将求解 N-S 方程的 Newton 迭代法、Oseen 迭代法和 Stokes 迭代法与上述完全重叠型区域分解技巧相结合, 研究了三种并行有限元迭代算法. 理论分析与数值实验表明, 通过粗细网格尺寸的适当选取, 这些并行迭代算法能取得渐近最优收敛阶的有限元解. 表 4 给出了准确解为

表 4 基于完全重叠型区域分解并行 Newton 迭代法数值结果 [44]

h	H	m	$\frac{\ \nabla(\mathbf{u}-\mathbf{u}_m^h)\ _{0,\Omega}}{\ \nabla\mathbf{u}\ _{0,\Omega}}$	$\frac{\ p-p_m^h\ _{0,\Omega}}{\ p\ _{0,\Omega}}$	$\mathbf{u}_{H^1}$ -收敛率	p_{L^2} -收敛率
1/16	1/4	2	0.2029967	0.000928442	—	—
1/36	1/6	2	0.0720316	0.000194599	1.27765	1.92688
1/64	1/8	2	0.0385427	7.94502e-05	1.08685	1.55695
1/100	1/10	2	0.0245793	4.32881e-05	1.00801	1.36068

$$\begin{aligned}
 u_1 &= x^2(x-1)^2y(y-1)(2y-1), \\
 u_2 &= -y^2(y-1)^2x(x-1)(2x-1), \\
 p &= 3x^2 + 3y^2 - 2,
 \end{aligned}
 \tag{4.1}$$

使用 Mini 元、4 个子区域时并行 Newton 迭代法的计算结果.

图 12 刻画了基于完全重叠型区域分解定常 N-S 方程与非定常 N-S 方程并行算法加速比随处理

图 11 二维两个子区域情形的完全重叠型区域分解

图 12 基于完全重叠型区域分解并行算法加速比

器个数的变化情况, 从中可以看出并行算法具有良好的并行性能. 对于非定常问题, 我们首先对空间施行完全重叠区域分解, 各个处理器在各自所负责的子区域使用向后 Euler 格式独立并行求解关于时间 t 的常微分方程, 对于非线性的对流项, 分别采用全隐格式和基于 Newton 迭代法与 Oseen 迭代法的半隐格式进行处理, 时间步长为 0.0001, $T = 0.01$.

将求解高 Reynolds 数流动的亏量 - 校正 (Defect-correction) 法与上述完全重叠型区域分解技巧相结合, 我们还研究了 N-S 方程的并行亏量 - 校正法. 该方法中, 每个处理器在局部加密的全局多尺度网格上使用亏量 - 校正方法独立并行地求解相应的子问题. 该方法能在较粗的网格上并行模拟高 Reynolds 数不可压缩流动问题. 图 13 就后台阶问题, 5 个子区域时我们的数值结果与文献 [46] 的数据进行了比较, 而图 14 和 15 则分别刻画了 $Re = 800$ 和 $Re = 1200$ 时我们的并行亏量 - 校正法计

图 13 并行亏量 - 校正法后台阶问题 $Re = 800$ 时的数值结果比较

图 14 并行亏量 - 校正法后台阶问题 $Re = 800$ (a) 和 $Re = 1200$ (b) 时速度场流线

图 15 并行亏量 - 校正法后台阶 $Re = 800$ (a) 和 $Re = 1200$ (b) 时压力等高线

算所得的速度场流线与压力等高线. 该数值结果验证了并行算法的有效性.

5 结束语

N-S 方程是描述流体运动的一类典型的非线性发展方程, 它出现于许多理论与应用研究领域, 如航天航空学、地球物理科学、水动力学、石油工业和等离子体物理等. 特别地, 其研究对人们认识和掌握流体的运动规律至关重要. 本文中, 我们设计了数值求解 N-S 方程的若干高效并行算法. 这些并行算法基于区域分解和两重网格离散方法, 使用有限元方法进行离散; 实现简单, 稍加修改现有的串行程序即可实现相应的并行计算; 通信需求少, 具有良好的并行性能. 理论分析与数值实验验证了算法的高效性. 进一步的研究方向包括非定常高 Reynolds 数不可压缩流动的全离散有限元并行算法, 以及高维复杂流动的大规模数值模拟等.

致谢 感谢审稿人提出的宝贵修改意见.

参考文献

- Shang Y Q. A two-level subgrid stabilized Oseen iterative method for the steady Navier-Stokes equations. J Comput Phys, 2013, 233: 210–226
- 阎超, 于剑, 徐晶磊, 等. CFD 模拟方法的发展成就与展望. 力学进展, 2011, 41: 562–589
- 钱战森, 张劲柏, 李椿萱. 基于预处理的不可压缩 N-S 方程拟压缩数值算法研究. 中国科学: 物理学 力学 天文学, 2010, 40: 1542–1554
- Behara S, Mittal S. Parallel finite element computation of incompressible flows. Parallel Comput, 2009, 35: 195–212
- 郭本瑜. 不可压缩粘性流问题的数值方法. 中国科学 A 辑, 1977, 20: 46–59
- Shang Y Q, Luo Z D. A parallel two-level finite element method for the Navier-Stokes equations. Appl Math Mech, 2010, 31: 1429–1438
- He Y N, Mei L Q, Shang Y Q, et al. Newton iterative parallel finite element algorithm for the steady Navier-Stokes equations. J Sci Comput, 2010, 44: 92–106
- 夏健, 田书玲, 王江峰, 等. 三维动态非结构重叠网格 Navier-Stokes 方程并行算法. 航空学报, 2008, 29: 1118–1124
- 李雪松, 徐建中. 一种非定常 N-S 方程并行求解设计. 工程热物理学报, 2008, 29: 52–54
- 李政, 金先龙, 元文果. 流体 - 结构耦合问题的有限元并行计算研究. 计算力学学报, 2007, 24: 727–732
- 鞠立力, 张林波. 解非定常不可压 Navier-Stokes 方程的压力修正投影法的并行算法研究. 计算数学, 1998, 20: 325–336
- Elman H, Howle V E, Shadid J, et al. A taxonomy and comparison of parallel block multi-level preconditioners for the incompressible Navier-Stokes equations. J Comput Phys, 2008, 227: 1790–1808
- Hwang F N, Cai X C. A class of parallel two-level nonlinear Schwarz preconditioned inexact Newton algorithms. Comput Methods Appl Mech Engrg, 2007, 196: 1603–1611
- Xia G H, Zhao Y, Yeo J H. Parallel unstructured multigrid simulation of 3D unsteady flows and fluid-structure interaction in mechanical heart valve using immersed membrane method. Comput & Fluids, 2009, 38: 71–79

- 15 Aulisa E, Manservigi S, Seshaiyer P. A computational multilevel approach for solving 2D Navier-Stokes equations over non-matching grids. *Comput Methods Appl Mech Engrg*, 2006, 195: 4604–4616
- 16 Tai C H, Zhao Y, Liew K M. Parallel-multigrid computation of unsteady incompressible viscous flows using a matrix-free implicit method and high-resolution characteristics-based scheme. *Comput Methods Appl Mech Engrg*, 2005, 194: 3949–3983
- 17 Sheu T W H, Wang M M T, Tsai S F. Element-by-element parallel computation of incompressible Navier-Stokes in three dimensions. *SIAM J Sci Comput*, 2000, 21: 1387–1400
- 18 Elias R N, Martins M A D, Coutinho A L G A. Parallel edge-based solution of viscoplastic flows with the SUPG/PSPG formulation. *Comput Mech*, 2006, 38: 365–381
- 19 Fujisawa T, Ito S, Inaba M, et al. Node-based parallel computing of three-dimensional incompressible flows using the free mesh method. *Eng Anal Bound Elem*, 2004, 28: 425–441
- 20 Sengupta T K, Dipankar A, Rao A. A new compact scheme for parallel computing using domain decomposition. *J Comput Phys*, 2007, 220: 654–677
- 21 Hwang F N, Cai X C. A class of parallel two-level nonlinear Schwarz preconditioned inexact Newton algorithms. *Comput Methods Appl Mech Engrg*, 2007, 196: 1603–1611
- 22 Nool M, Proot M M J. A parallel least-squares spectral element solver for incompressible flow problems on unstructured grids. *Parallel Comput*, 2005, 31: 414–438
- 23 Heinrichs W. Least-squares spectral collocation with the overlapping Schwarz method for the incompressible Navier-Stokes equations. *Numer Algorithms*, 2006, 43: 61–73
- 24 Turner D Z, Nakshatrala K B, Hjelmstad K D. A variational multiscale Newton-Schur approach for the incompressible Navier-Stokes equations. *Internat J Numer Methods Fluids*, 2010, 62: 119–137
- 25 Jiang B. A parallel domain decomposition method for coupling of surface and groundwater flows. *Comput Methods Appl Mech Engrg*, 2009, 198: 947–957
- 26 Rebollo T C, Vera E C. Study of a non-overlapping domain decomposition method: Steady Navier-Stokes equations. *Appl Numer Math*, 2005, 55: 100–124
- 27 Choi H, Kang S, Yoo J Y. Parallel large eddy simulation of turbulent flow around MIRA model using linear equal-order finite element method. *Internat J Numer Methods Fluids*, 2008, 56: 823–843
- 28 Ma Y C, Zhang Z M, Ren C F. Local and parallel finite element algorithms based on two-grid discretization for the stream function form of Navier-Stokes equations. *Appl Math Comput*, 2006, 175: 786–813
- 29 Shang Y Q, Wang K. Local and parallel finite element algorithms based on two-grid discretizations for the transient Stokes equations. *Numer Algorithms*, 2010, 54: 195–218
- 30 Shang Y Q, He Y N. Parallel finite element algorithm based on full domain partition for stationary Stokes equations. *Appl Math Mech*, 2010, 31: 643–650
- 31 Dolean V, Lanteri S. Parallel multigrid methods for the calculation of unsteady flows on unstructured grids: algorithmic aspects and parallel performances on clusters of PCs. *Parallel Comput*, 2004, 30: 503–525
- 32 Aulisa E, Cervone A, Manservigi S, et al. A multilevel domain decomposition approach for studying coupled flow applications. *Commun Comput Phys*, 2009, 6: 319–341
- 33 莫则尧, 李晓梅, 张宝琳. 并行多重网格算法求解高 Reynolds 数不可压 Navier-Stokes 方程. *数值计算与计算机应用*, 1998, 19: 135–143
- 34 Udaykumar H S, Krishnan S, Marella S V. Adaptively refined, parallelised sharp interface Cartesian grid method for three-dimensional moving boundary problems. *Int J Comput Fluid Dyn*, 2009, 23: 1–24
- 35 Santiago E D, Law K H. A distributed implementation of an adaptive finite element method for fluid problems. *Comput Structures*, 2000, 74: 97–119
- 36 周春华. 流动数值模拟中一种并行自适应有限元算法. *计算物理*, 2006, 23: 412–418
- 37 Shang Y Q, He Y N. A parallel Oseen-linearized algorithm for the stationary Navier-Stokes equations. *Comput Methods Appl Mech Engrg*, 2012, 209–212: 172–183
- 38 He Y N, Xu J C, Zhou A H. Local and parallel finite element algorithms for the Navier-Stokes problem. *J Comput Math*, 2006, 24: 227–238
- 39 Ma F Y, Ma Y C, Wo W F. Local and parallel finite element algorithms based on two-grid discretization for steady Navier-Stokes equations. *Appl Math Mech*, 2007, 28: 27–35
- 40 Shang Y Q, He Y N, Luo Z D. A comparison of three kinds of local and parallel finite element algorithms based on two-grid discretizations for the stationary Navier-Stokes equations. *Comput Fluids*, 2011, 40: 249–257
- 41 Shang Y Q. A parallel two-level linearization method for incompressible flow problems. *Appl Math Lett*, 2011, 24: 364–369

- 42 Shang Y Q, He Y N, Kim D W, et al. A new parallel finite element algorithm for the stationary Navier-Stokes equations. *Finite Elem Anal Des*, 2011, 47: 1262–1279
- 43 Ghia U, Ghia K N, Shin C T. High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method. *J Comput Phys*, 1982, 48: 387–411
- 44 Shang Y Q, He Y N. Parallel iterative finite element algorithms based on full domain partition for the stationary Navier-Stokes equations. *Appl Numer Math*, 2010, 60: 719–737
- 45 尚月强, 何银年. 非定常 Navier-Stokes 方程基于完全重叠型区域分解的有限元并行算法. *计算物理*, 2011, 28: 181–187
- 46 Gartling D K. A test problem for outflow boundary conditions-flow over a backward-facing step. *Internat J Numer Methods Fluids*, 1990, 11: 953–967

Parallel numerical methods for incompressible flows

SHANG YueQiang & HE YinNian

Abstract The simulation of incompressible flows consists of an important part of computational fluid dynamics. Based on finite element discretization, this paper develops several parallel numerical methods for the incompressible flows governed by the Navier-Stokes (N-S) equations, which can be classified into two classes: one is based on two-grid discretization, where the fully nonlinear N-S equations are first solved on a coarse grid, and the correction is then calculated on a fine grid by solving a linear residual problem in a parallel manner; the other is based on a novel fully overlapping domain decomposition technique, where each processor computes a local finite element solution in its own sub-domain using a global multiscale mesh that is locally refined around its own sub-domain. These parallel numerical methods are easy to implement. They have low communication complexity, and can yield a finite element solution with the same order of convergence rate as the standard finite element methods. Theoretical analysis and numerical tests demonstrated the high efficiency of the studied methods.

Keywords incompressible flows, Navier-Stokes equations, finite element, parallel algorithm, domain decomposition, two-grid method

MSC(2010) 76D15, 35Q14

doi: 10.1360/012012-37