

大连理工大学本科毕业设计（论文）

关于 Stokes 方程的预条件方法

Preconditioning Methods about Stokes Equations

学 院（系）： 数学科学学院

专 业： 数学与应用数学

学 生 姓 名： 李东洋

学 号： 201311005

指 导 教 师： 张晨松 李崇君

评 阅 教 师： 孟兆良

完 成 日 期： 2017. 6. 11

大连理工大学

Dalian University of Technology

摘 要

Stokes 方程是描述不可压缩流体运动的一类特殊方程. 该方程经过稳定有限元离散后便得到了线性方程组的鞍点问题. 当系数矩阵的阶数很大时, 如果使用直接法求解会花费很大的存储空间和计算时间, 所以我们经常使用间接法, 通常是 Krylov 子空间方法, 而其需要许多次的迭代. 因此求解该问题时还要使用预条件子来加速 Krylov 子空间迭代法的收敛.

在本文中, 我们先大致介绍了有限元离散理论及方法, 把原问题离散得到了具有特殊结构的鞍点系统. 然后介绍了 Krylov 子空间方法和多重网格法, 结合这两种方法将之运用到鞍点系统求解上. 本文正文综述了目今一些求解该问题的预条件子, 介绍了各个预条件子的由来, 以及它的理论知识, 并且给出了大部分预条件子的算法实现步骤, 然后通过数据实验分析来比较各个预条件子的表现情况.

本文的实验主要包括四个部分. 第一部分, 我们比较了不同的预条件子配合不同的迭代法, 发现了 FGMRES 方法相比于 MINRES 和 GMRES 有着更好的表现效果. 第二部分, 我们发现在不同大小的网格下, UZAWA 预条件子的迭代次数是最小的, 预条件效果最好, 并且具有很好的鲁棒性. 第三部分, 在 UZAWA 预条件子的算法中, 有一个参数 ω , 而后我们通过实验找到了 UZAWA 预条件子的最优参数 ω 的大致范围, 发现优秀的参数 ω 与网格单元的数目大致是一致的, 因此我们在使用 UZAWA 预条件子时可以根据网格单元的数目选取参数 ω 以得到更好的收敛效果. 第四部分, 我们发现 MG 方法作为预条件子也有不错的效果, W-型循环相比于 V-型循环鲁棒性更好.

关键词: Stokes 方程; 有限元方法; Krylov 子空间方法; 多重网格法; 预条件子

Preconditioning Methods about Stokes Equations

Abstract

The Stokes equation is a special kind of equation that describes the movement of incompressible fluids. When the order of the coefficient matrix is large, if the direct method is used to solve it, it will take a lot of storage space and computation time. Therefore, we often use the indirect method to solve the problem, Which is usually the Krylov subspace method, which requires many iterations, so the problem is solved by using the preconditioner to accelerate the convergence of the Krylov subspace iterative method.

In this paper, we first introduce the finite element discretization theory and method, the original problem is discrete with a special structure of the saddle point system. Then we introduce the Krylov subspace method and multigrid method, and combine these two methods to solve the saddle point system. The paper summarizes some preconditioners to solve the problem, introduces the origin of each preconditioner, and its theoretical knowledge, and gives the steps of the algorithm of most preconditioners, And then through the experimental analysis of data to compare the performance of each preconditioner.

The experiment of this paper mainly includes four parts. In the first part, we compare the different preconditioners with different iterations and find that the FGMRES method has better performance than MINRES and GMRES. In the second part, we find that under grids of different sizes, the number of iterations of UZAWA is the smallest, so the precondition is the best and the robustness is very good. Thirdly, in the UZAWA preconditioner algorithm, there is a parameter ω , and then through experiments , we find that the excellent parameter ω is roughly the same as the number of grid cells, so when we use UZAWA preconditioner we can select the parameter ω according to the number of grid cells to get better iteraions. The fourth part, we find that the MG method as a preconditioner also has a good effect, W-type cycle has a better robustness than V-type cycle.

Key Words: Stokes equations; finite element method; Krylov subspace iterative method; multigrid method; preconditioners

目 录

| | |
|--|----|
| 摘 要 | I |
| Abstract | II |
| 1 文献综述 | 1 |
| 1.1 Navier-Stokes 方程的背景介绍 | 1 |
| 1.2 国内外发展历程和研究状况 | 2 |
| 1.3 本文主要脉络 | 4 |
| 2 有限元方法, 子空间方法和 MG 方法 | 5 |
| 2.1 有限元方法 | 5 |
| 2.2 Krylov 子空间方法 | 7 |
| 2.3 多重网格法 | 8 |
| 3 求解 Stokes 方程的预条件子 | 10 |
| 3.1 块对角预条件子 | 10 |
| 3.2 块上三角预条件子 | 10 |
| 3.3 块下三角预条件子 | 11 |
| 3.4 块 LU 预条件子 | 11 |
| 3.5 SIMPLE 预条件子 | 12 |
| 3.6 SIMPLER 预条件子 | 12 |
| 3.7 MSIMPLER 预条件子 | 13 |
| 3.8 UZAWA 预条件子 | 13 |
| 3.9 投影预条件子 | 13 |
| 3.10 DGS 预条件子 | 14 |
| 3.11 LSC 预条件子 | 14 |
| 3.12 PMM 预条件子 | 15 |
| 3.13 AL 预条件子 | 15 |
| 4 数值实验 | 17 |
| 4.1 不同 Krylov 子空间方法配合各个预条件子的数值实验 | 18 |
| 4.2 不同的预条件子在不同网格下的数值实验 | 19 |
| 4.3 UZAWA 预条件子的最优参数选取数值实验 | 20 |
| 4.4 MG 预条件子的数值实验 | 21 |
| 结 论 | 23 |
| 1 本文的主要贡献 | 23 |

| | |
|-----------------|----|
| 2 下一步工作展望 | 23 |
| 参 考 文 献 | 24 |
| 致 谢 | 26 |

1 文献综述

1.1 Navier-Stokes 方程的背景介绍

Navier-Stokes 方程是描述粘性流体运动规律的关键控制方程, 更多理论知识参考文献[1].

Navier-Stokes 方程

$$\begin{cases} \frac{\partial \vec{u}}{\partial t} - \nu \Delta \vec{u} + (\vec{u} \cdot \nabla) \vec{u} + \nabla p = \vec{f}, & x \in \Omega, \\ \nabla \cdot \vec{u} = 0, & x \in \Omega. \end{cases} \quad (1.1)$$

其中, $\vec{u} = \vec{u}(x, t)$ 是速度场, ν 是运动学粘性常数, $p = p(x, t)$ 是压力, $\vec{f} = \vec{f}(x, t)$ 为外力, Ω 是空间维数为 d 的开区域. 配合边界条件及初始条件

$$\begin{cases} \vec{u} = \vec{g}, & x \in \partial\Omega_D, \\ \nu \frac{\partial \vec{u}}{\partial \vec{n}} - \vec{n}p = 0, & x \in \partial\Omega_N. \end{cases} \quad (1.2)$$

$$\vec{u}(x, 0) = \vec{u}_0(x), \quad x \in \Omega. \quad (1.3)$$

其中 $\partial\Omega_D$ 是 Dirichlet 边界, $\partial\Omega_N$ 是 Neuman 边界.

该方程组实际上就是质量守恒定律, 动量定理以及牛顿第二定律在流体力学中的表达方式.

Navier-Stokes 方程在不依赖时间 t , 即去除 $\frac{\partial \vec{u}}{\partial t}$ 项, 可得到定常流方程:

$$\begin{cases} -\nu \Delta \vec{u} + (\vec{u} \cdot \nabla) \vec{u} + \nabla p = \vec{f}, & x \in \Omega, \\ \nabla \cdot \vec{u} = 0, & x \in \Omega. \end{cases} \quad (1.4)$$

Stokes 方程是 Navier-Stokes 方程的一种, 在速度比较小时, 去除非线性部分后, 即去除对流项 $(\vec{u} \cdot \nabla) \vec{u}$ 就能得到 Stokes 方程:

$$\begin{cases} \frac{\partial \vec{u}}{\partial t} - \nu \Delta \vec{u} + \nabla p = \vec{f}, & x \in \Omega, \\ \nabla \cdot \vec{u} = 0, & x \in \Omega. \end{cases} \quad (1.5)$$

Stokes 方程在不依赖时间 t , 即去除 $\frac{\partial \vec{u}}{\partial t}$ 项, 可得到其定常流方程:

$$\begin{cases} -\nu \Delta \vec{u} + \nabla p = \vec{f}, & x \in \Omega, \\ \nabla \cdot \vec{u} = 0, & x \in \Omega. \end{cases} \quad (1.6)$$

采用 Picard 或 Newton 固定迭代隐式时间离散方法, 线性化 Navier-Stokes 方程即可得到 Ossen 方程

$$\begin{cases} \alpha \vec{u} - \nu \Delta \vec{u} + (\vec{v} \cdot \nabla) \vec{u} + \nabla p = \vec{f}, & x \in \Omega, \\ \nabla \cdot \vec{u} = 0, & x \in \Omega. \end{cases} \quad (1.7)$$

这里 \vec{v} 是上一步未知速度, α 是离散时间间隔的倒数, 当 $\vec{v} = 0$ 时就是 Stokes 问题.

Navier-Stokes 方程已经在许多领域得到了广泛的运用, 如飞行器制造, 船舶运动, 石油挖掘, 医学方面的心脑血管治疗等, 并且研究该方程对掌握流体的运动规律十分重要. 但由于该方程的非线性性和耦合性使得该方程的精确解很难找到, 因此不得不求助于数值模拟以了解解的性质和状态. 由于流体所在的区域复杂, 并且在不同的地方和不同的时间段表现出不同的物理特征, 再加上它的高度震荡性, 使得数值模拟很难成功, 需要更复杂的网格来模拟, 也因此矩阵规模比较大^[2]. 所以, 求解该方程就归结于求大规模矩阵的问题.

1.2 国内外发展历程和研究状况

1.2.1 预条件方法发展历程

大型稀疏线性方程组系统的求解一直是计算科学研究的热点. 传统的解决方法主要分为两类, 直接法和迭代法. 但这种分类过于简单, 一则两类方法的界限越来越不清晰. 再则从经典 Jacobi 方法, Gauss-Seidel 和 SOR 迭代, 再到在理论上必定几步内必收敛的 Krylov 子空间方法, 再到多层迭代法, 迭代法越来越多, 越来越值得依赖. Trefethen 和 Bauapty 说, 在计算机科学中没有什么比把一个看起来特别棘手的矩阵求解问题转换成一个求解起来逼近非常迅速的问题更有艺术性的事了^[3]. 对于 Krylov 子空间方法, 就是预条件法. 实际上研究有效的预条件子已经比研究单纯的直接法或者是 Krylov 子空间方法更是火热了. 而且现在并没有一种更通用的预条件技巧以适用所有情况, 至少在可预见的未来是这样的^[3]. 预条件就是把要求解的系统转变成另一个系统, 这个系统对于迭代求解的话有非常好的性质, 这样一来解原问题就变得简单许多. 预条件就是这个转变所用的矩阵, 它其实是要改变系数矩阵的谱, 以改变迭代算法的收敛条件和收敛速度. 预条件是在 1948 年由 Turing 在他的关于直接法的舍入误差的影响的论文中第一次提出的, 而第一次跟迭代法相联系则是在 1996 年 Evans 的 SSOR 的切比雪夫加速化的论文中. 然而预条件用于提高迭代算法的收敛性最远可追溯到十九世纪的 Jacobi. 预条件作为减小条件数的一种方法则是 1937 年由 Cesari 提出的. 还有一些早期的对共轭梯度法 (CG) 的预条件可以在 1960 年代的一些论文中找到. SSOR 预条件对于加速共轭梯度法的详细研究则是 1972 年由 Axelsson 发表. 在 1970 年代 Meijerink 和 van der Vorst 的不完全 Cholesky 共轭梯度算法 (ICCG) 带来了很大突破. 不完全矩阵分解是 Bulev 和 Vargr 单独提出的, 然而却是 Meijerink 和 van der Vorst 认识到了不完全矩阵分解作为共轭梯度法的预条件的潜力, Kershaw 的论文则是使这一方法受欢迎起来. 之后的这些年预条件方法更是得到发展, 包括稳定化, 块矩阵, 层填充, 甚至是并行实现等^[4].

1.2.2 预条件方法应用在 Stokes 方程的现状

对于不可压缩的 Navier-Stokes 方程的数值求解, 预条件法取得了很好的效果. 近些年计算机软件技术的快速发展使得解决 Navier-Stokes 方程的数值方法得到更好的发展, 如有限元法, 有限差分法, 有限体积法等用于方程的离散化, krylov 子空间方法等用于求解大型稀疏线性方程组. 早些年这类问题的关注点在于如何得到好的离散化, 现在则是关注于能否得到有效解. 运用隐式时间的有限元方法离散和 Picard (或 Newton) 线性化后^[2,3], 我们得到具有如下矩阵结构的大型稀疏线性系统

$$\begin{pmatrix} A & B^T \\ B & -C \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}. \quad (1.8)$$

一般那情况下, $A \in \mathbb{R}^{m \times m}$ 是对称正定的, $B \in \mathbb{R}^{m \times n}$ 是列满秩的, $C \in \mathbb{R}^{n \times n}$ 是半正定的. 当 $C = 0$ 时, 即

$$\begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}. \quad (1.9)$$

这时候方程组便被称作鞍点系统. 解决鞍点系统已有许多方法, 如直接法, 预条件后的 Krylov 子空间方法, MINRES 算法, Uzawa 方法等. 直接解这类方程通常需要编序的技巧以避免零主元^[4]. 一种方式就是分离压力项和速度项的计算步骤, 例如有种方法叫做惩罚函数法, 但由于惩罚参数的存在, 对线性方程系统的条件要求很高, 迭代解的求出是困难的. 还有一种方法就是用 Uzawa 型迭代方案^[2, 5, 6], 但是不幸的是收敛缓慢. 近数十年来一些针对智能编序和经典的迭代算法结合的方法被提出来解决鞍点系统, 如 Wille 提出的 SILU 和 ILU 方案^[4]. 还有一些方法则是基于分离的. 方程系统被分为速度项和压力项. 整个系统由一种迭代法求解, 如 Krylov 子空间方法, 由于其一般不会破坏系数矩阵的稀疏性, 只需要重复计算矩阵与向量的乘积, 但是求解时也会收敛的很慢. 有两种方法可以利用, 一种是多重网格法, 还有一种就是预条件 Krylov 子空间方法.

近二十年来许多学者做了大量工作, 提出来了各种预条件子, 以提高 Krylov 子空间方法求解 Stokes 方程的收敛速度. 一种经常用的预条件技巧是基于系数矩阵的 LU 分解, 如块预条件子, (块)三角和对角预条件子. 也有采用分层策略的预条件子, 这些方法关键是找到一个合适的 Schur 补 $BA^{-1}B^T$. 我们已经有了有一些比较好的经典的块预条件子^[3,7], 例如 Kay, Logan 和 Wathen 的压力对流扩散对流换位子 (PCD), Elman, Howle, Shadid, Shuttleworth 和 Tuminaro 的最小二乘换位子 (LSC), Benzi 和 Olshanskii 的增广拉格朗日方法 (AL)^[8], 人工压缩性 (AC) 预条件子和梯度-散度 (GD) 预条件子, HSS 预条件子, 维数分裂 (DS) 预条件子, 松弛化维数因式分解 (RDF) 预条件子^[2,5]. SIMPLE 类型的预条件子组成了许多不同的预条件子, 它们其实也可以看作块预条件子^[9]. 除了标准的预条件子, 还有 SIMPLER 等其它变式的预条件子^[3]. 本文中主要说明一些预条件

子:块对角预条件子, 块下三角预条件子, 块上三角预条件子, 块 LU 预条件子, SIMPLE 预条件子, SIMPLER 预条件子, MSIMPLER 预条件子, UZAWA 预条件子, DGS 预条件子, LSC 预条件子, PMM 预条件子, AL 预条件子.

1.3 本文主要脉络

本文主要研究求解 Stokes 方程所用到的一些预处理方法, 大致内容如下:

第一部分, 介绍有限元方法的基本知识, 着重介绍了求解 Stokes 方程组得有限元空间; 介绍 Krylov 子空间方法的几种常用方法; 又介绍多重网格方法, 如 GMG 算法; 第二部分, 介绍求解 Stokes 方程常用的诸多预条件方法以及其算法理论基础; 第三部分则是数值实验部分, 本文试图通过结合 MG 和 Krylov 子空间方法等, 对比这些预条件子的数值结果, 来说明各个预条件子的表现状况, 重点介绍了表现比较好的 UZAWA 预条件子以及其参数 ω 的选取, 最后做出总结并对下一步做出展望.

2 有限元方法, 子空间方法和 MG 方法

2.1 有限元方法

考虑 Stokes 方程 (1.5), 由于这个方程依赖于时间项, 我们在进行时间离散后, 可以得到类似于 Stokes 方程的形式 (1.7). 我们进一步简化该问题, 即讨论定常流下的 Stokes 问题

$$\begin{cases} -\Delta \vec{u} + \nabla p = \vec{f}, & x \in \Omega, \\ \nabla \cdot \vec{u} = 0, & x \in \Omega. \end{cases} \quad (2.1)$$

令 $V := [H_0^1(\Omega)]^d$, $Q := L_0^2(\Omega) := \{q \in L^2(\Omega) : \int_{\Omega} q dx = 0\}$, 通过变分方法我们得到弱形式

$$\begin{cases} 2 \int_{\Omega} \varepsilon(u) : \varepsilon(v) dx + \int_{\Omega} p \nabla \cdot v dx = \int_{\Omega} f v dx, \quad \forall v \in V, \\ \int_{\Omega} q \nabla \cdot u dx = 0, \quad \forall q \in Q. \end{cases} \quad (2.2)$$

V 是满足本质边界条件的 Soblev 空间, 式 (2.2) 中的 $:$ 表示并向量积. 事实上这是个混合元问题, 有些方法如惩罚法就是从这里出发的^[10].

记 $a(u, v) = \int_{\Omega} \varepsilon(u) : \varepsilon(v) dx$, $b(v, q) = \int_{\Omega} q \nabla \cdot v dx$, V' 和 Q' 分别是 V 和 Q 的对偶空间. 那么上面的问题也就可以写成:

$$\begin{cases} a(u, v) + b(v, q) = \langle f, v \rangle, \quad \forall v \in V, \\ b(u, q) = 0, \quad \forall q \in Q. \end{cases} \quad (2.3)$$

这里 $a(\cdot, \cdot) : V \times V \rightarrow \mathbb{R}$, $b(\cdot, \cdot) : V \times Q \rightarrow \mathbb{R}$, 是连续型的双线性形式, 也就是

$$a(u, v) \leq C_a \|u\|_V \|v\|_V, \quad \forall u, v \in V, \quad b(u, p) \leq C_b \|u\|_V \|p\|_Q, \quad \forall u \in V, p \in Q.$$

定义线性算子 $A : V \rightarrow V'$, 使得 $\langle Au, v \rangle = a(u, v)$, $\forall u, v \in V$, 线性算子 $B : V \rightarrow Q'$, 使得 $\langle Bu, p \rangle = \langle u, B^T p \rangle = b(u, p)$, $\forall u \in V, p \in Q$.

这样我们就可以把问题写作算子形式, 得到的便是著名的鞍点问题:

$$\begin{cases} Au + B^T p = f, \\ Bu = g. \end{cases} \quad (2.4)$$

下面我们说明 Stokes 问题是适定性的^[11]. 适定性是指: 解是存在的; 解是唯一的; 解连续依靠于初始条件, 即给予初始条件比较小的扰动, 解的变化也是比较小的. 使用数值算法近似求解问题的非常关键一点就是问题是否是适定的. 因为如果前两个条件是满足的, 但该解如果没有对初始条件的连续依赖性, 那么逼近精确解很可能是没有价值的, 这是因为在近似求解的时候给出的初始条件的一点扰动 (比如机器精度带来的偏差) 对结果的扰动可能造成巨大的影响, 那么近似求出来的解就和原来的解相差就很远了, 这样的解是没有意义的.

很容易证明 $\text{Im } B = L^2(\Omega)/R = Q$, 再利用 Brezzi 定理^[11, 12], 我们就得到了以下定理: 对于 Stokes 方程的弱形式 (1. 2), 存在唯一的解 $(u, p) \in V \times Q$, 并且 $\|u\|_V + \|p\|_Q \leq C \|f\|_{H^{-1}}$. 这样就说明了 Stokes 方程是适定的.

现在选择一种有限元空间逼近 $V_h \subset V$ 和 $Q_h \subset Q$, 我们得到离散类型的问题:

$$\begin{cases} 2 \int_{\Omega} \varepsilon(u_h) : \varepsilon(v_h) dx + \int_{\Omega} p \nabla \cdot v_h dx = \int_{\Omega} f v_h dx, \quad \forall v_h \in V_h, \\ \int_{\Omega} q \nabla \cdot u_h dx = 0, \quad \forall q_h \in Q_h. \end{cases} \quad (2.5)$$

u_h 和 p_h 各自代表离散后的速度和压强. 方程解的存在性是不难验证的, 但由于在方程 (2. 4) 的第二个式中缺少压力项, 方程的未知项可能是任意的速度和压力的组合. 因此为了保证有限元逼近解的唯一性, 我们通常要求离散的有限元空间满足 inf-sup 条件:

$$\inf_{q_k \in Q_k} \sup_{v_k \in V_k} \frac{(\nabla \cdot v_k, q_k)}{\|v_k\|_{V_k} \|q_k\|_{Q_k}} \geq k_h \geq k_0 > 0. \quad (2.6)$$

有限元离散后满足 Ladyshenskaya-Brezzi-Babuska (LBB) 条件是指

$$\inf_{q_k \in Q_k} \sup_{v_k \in V_k} \frac{(\nabla \cdot v_k, q_k)}{\|v_k\|_{V_k} \|q_k\|_{Q_k}} \geq \gamma > 0. \quad (2.7)$$

如果一种有限元是不满足 LBB 条件的, 运用稳定化技巧的话, 方程 (2. 4) 的解才有可能得到.

我们把离散得到的方程写作矩阵形式便是

$$\begin{pmatrix} A & B^T \\ B & -C \end{pmatrix} \begin{pmatrix} u_h \\ p_h \end{pmatrix} = \begin{pmatrix} f_h \\ g_h \end{pmatrix}. \quad (2.8)$$

$A \in \mathbb{R}^{n \times n}$ 是扩散, 对流和时间依赖项的离散化, 矩阵 A 也关系到稳定项, 矩阵 $B \in \mathbb{R}^{m \times n}$ 和 $B^T \in \mathbb{R}^{n \times m}$ 是 (负) 离散散度和梯度. A 通常是对称正定的, C 通常对称半负定. 如果是 LBB 稳定的有限元对, 不需要压力稳定化, 也就是说 $C = 0$ ^[12], 也就是常见的鞍点问题.

接下来我们介绍二维空间下的几种常见的有限元对^[10]. 比如 $P_1 - P_1$ 元, $P_1 - P_0$ 元, 这些有限元通常不满足 inf-sup 条件, 因此并不用于求解 Stokes 方程. $P_1 - P_1$ 元指的是这时有限元空间 $V_h = L_1^1 \cap V$, $Q_h = L_1^1 \cap Q$. 类似的可以定义 $P_1 - P_1$ 元. $Q_1 - Q_1$ 元则是矩形网格上的有限元. 使用上面的两个有限元的话就会很快意识到它们的结果严重依赖于网格, 解出来的压力也很奇怪地不稳定. $P_1 - P_0$ 元则是可以想象的最简单的逼近不可压缩流体的有限元了. 由于 P_1 速度场的散度是分块常数的, 可以得知是近似零散度的, 并且 $O(h)$ 的近似更加平衡, 可是也很容易知道这类元对于网格不具有普遍性.

$P_2 - P_0$ 元是指连续的分块二次向量逼近速度项, 分块常数逼近压力项. 可以看到由于这个有限元对不平衡地逼近, 对 V_h 是 $O(h^2)$ 的逼近, 而对 Q_h 只有 $O(h)$ 的逼近. 我们可

以证明 $P_2 - P_0$ 元满足 inf-sup 条件, 也就是说这个有限元对具有很好的稳定性. 类似的还可以证明 $Q_2 - P_0$ 元 (Q 则指的是矩形单元) 也是满足的.

Hood-Taylor 元, 主要指 $P_k - P_{k-1}$ 元, $Q_k - Q_{k-1}$ 元. 这里我们讨论前者. Hood 和 Taylor 已经证明 $P_k - P_{k-1}$ 元在 $k \geq 2$ 时满足 inf-sup 条件当且仅当每个 Γ_h 至少含有三个三角形, Γ_h 是 Ω 的一个常规的三角化序列^[7]. 也就是说, 我们可以通过好的网格划分得到稳定的 $P_2 - P_1$ 元, 后面我们将和 $P_2 - P_0$ 元做数值比较, 来观察它们两者的表现. 求解 Stokes 方程还有其他的稳定的有限元对, 比如 MINI 元, Crouzeix-Raviart 元等, 这里就不做比较了.

2.2 Krylov 子空间方法

考虑下面的大型线性方程组

$$Ax = b, A \in \mathbb{R}^{n \times n}, b \in \mathbb{R}^n, \quad (2.9)$$

其中 $x \in \mathbb{R}^n$ 是要所求的解. 目前求解此类系统的首选方法就是 Krylov 子空间算法^[13]. 其基本思想是在子空间 $K \subset \mathbb{R}^n$ 中找到满足条件的近似解.

设 $A \in \mathbb{R}^{n \times n}, r \in \mathbb{R}^n$, 我们称

$$K_m(A, r) = \text{span}\{r, Ar, \dots, A^{m-1}r\} \subset \mathbb{R}^n$$

是由 A 和 r 生成的 Krylov 子空间.

设 $\dim(K) = m$, 我们要求残量满足 m 个正交性条件: $r = b - A\tilde{x} \perp L$. 其中 \tilde{x} 是要找的近似解, L 是另一个 m 维子空间, 称 L 为约束子空间.

对于如何选取约束子空间 L , 通常有以下几种:

$L=K$, 也就是满足 Galerkin 条件, 如 FOM (完全正交法), CG (共轭梯度法).

$L=AK$, 如 MINRES (极小残量法), GMRES (广义极小残量法), FGMRES (可变广义极小残量法).

$L=K(A^T, r)$, 如 BiGG (双共轭梯度法).

预条件共轭梯度法 (PCG): CG 的收敛速度取决于系数矩阵的条件数, 条件数越小, 收敛速度越快. 为了提高一些病态问题的收敛速度, PCG 法被提出来了, 好的预条件可以更好的改变系数矩阵的谱分布, 大大加快收敛速度.

MINRES, GMRES, FGMRES 方法后者都是对前者的改进, 这里我们大致介绍 GMRES 算法, 具体可参考^[13-15], 步骤如下:

- ①给出初始问题, 初始解以及子空间维数, 并确定收敛准则;
- ②运用 Arnoldi 过程以生成 Krylov 子空间的基;
- ③求解最小二乘问题, 然后得到最小二乘解;

④用求来的最小二乘解和子空间的基向量得到方程的近似解。

2.3 多重网格法

多重网格 (MG) 法是利用离散后网格的继承性来求解微分方程的一组算法。它的主要思想是通过时时的全局校正 (细网格负责消除高频误差, 粗网格消除低频误差), 最后近似求解粗网格问题, 这个粗网格问题和细网格问题类似但是求解起来更容易。其实, MG 方法也可以作为一种预条件方法, 但是对于大规模工程计算, 该方法将会作为一种更普遍自然的思想。该方法的主要步骤如下^[2]:

①松弛 (磨光): 通过几次的磨光步骤消除高频误差。磨光算子通常是一些迭代法, 比如常见的 GS, SOR, CG 等。

②限制: 将细网格的残量限制到更粗层的网格, 一直进行下去, 直到需要的层即为最粗的层。

③粗网格校正: 在最粗的网格求解近似问题, 得到最初层的解。

④修正: 将在最粗网格计算得到的解投到更细网格上, 得到较粗层上的方程近似解, 一直进行, 直到到达最细层, 得到的解即为最终解。

图 2.1 形象地介绍了多重网格法的过程。

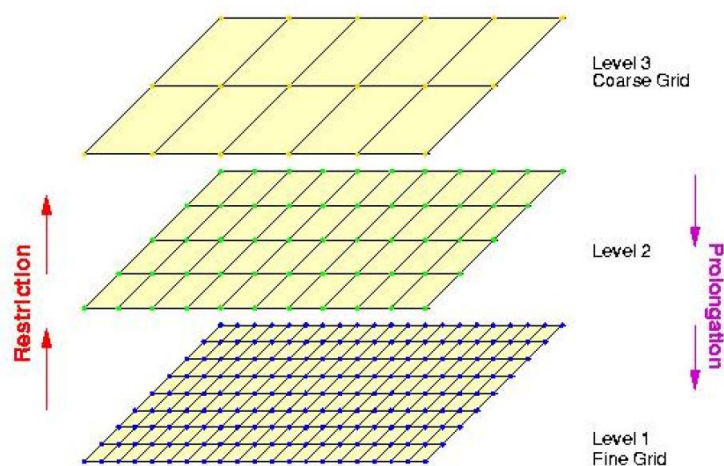


图 2.1 多重网格方法

GMG (几何多重网格) 方法主要有 V-型, W-型, 全网格几种类型。在这里我们大致说明 V-型循环和 W-型循环的每次迭代过程。

V-型循环的一次迭代:

①前磨光: 对 $k=1, 2, 3 \dots v \leftarrow v + S_l(r_l - A_l v)$;

②粗网格校正：通过迭代算子 B_{l-1} 找到在 $l-1$ 层的残量方程的近似解 e_{l-1} , $P_{l,l-1}$ 是延伸矩阵（将粗网格的变化到细网格的矩阵）, $I_{l,l-1}$ 是限制矩阵（将细网格的变化到粗网格的矩阵）. $e_{l-1} \leftarrow B_{l-1} I_{l,l-1} (r_l - A_l v)$, $v \leftarrow v + P_{l-1,l} e_{l-1}$;

③后磨光: $v \leftarrow v + S_l^T (r_l - A_l v)$.

W-型循环的一次迭代:

①前磨光: 对 $k = 1, 2, 3 \dots v \leftarrow v + S_l (r_l - A_l v)$;

②粗网格校正：通过迭代算子 B_{l-1} 找到在 $l-1$ 层的残量方程的近似解 e_{l-1} , $P_{l,l-1}$ 是延伸矩阵, $I_{l,l-1}$ 是限制矩阵. $e_{l-1} \leftarrow B_{l-1} (I_{l,l-1} (r_l - A_l v) - A_{l-1} e_{l-1})$; $v \leftarrow v + P_{l-1,l} e_{l-1}$;

③后磨光: $v \leftarrow v + S_l^T (r_l - A_l v)$.

图 2.2 大致对比了一下它们的区别, 并且在后面的实验里对比它们作为预条件子的表现状况.

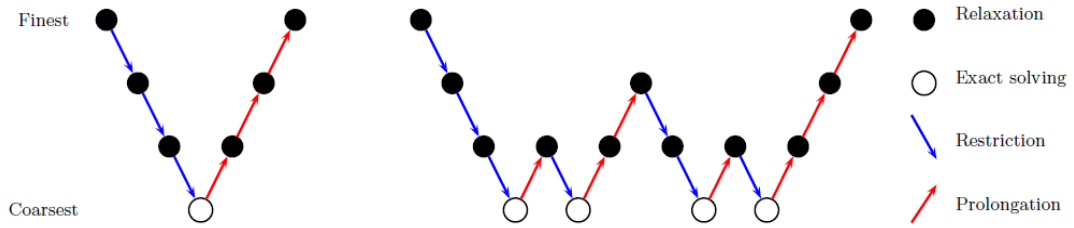


图 2.2 V-型循环和 W-循环示意图

常规网格中的各向异性系统的问题, 或各向同性系统但在各向异性网格的问题, 会使得几何多重网格方法变得不适用起来. 问题的关键就是为了能用 GMG 方法, 必须要事先知道网格是什么样的. 为了消除这些麻烦因此便有了 AMG (代数多重网格) 方法. AMG 和 GMG 方法一样也有上面说的初始化, 磨光, 修正, 限制等, 不一样的是, AMG 方法最初并不需要一个固定的网格, 而是数据点会根据随机模型在后面生成. 未知点也有彼此的继承关系, 不过事先并不知道最初的位置. AMG 方法是假定给出一个简单的松弛过程 (通常是逐点 Gauss-Seidel 松弛), 然后尝试使用系数矩阵 A 的代数信息构建一个合适的运算符相关内插. AMG 方法通常有经典 AMG 方法, 基于聚合的 AMG 方法, 它又包括 UA AMG (非磨光的基于聚合 AMG 方法) 和 SA AMG (磨光的基于聚合 AMG 方法). 本文对预条件子的实验基本使用经典 AMG 方法^[8].

3 求解 Stokes 方程的预条件子

本节将详细介绍一些求解 Stokes 方程的预条件子, 给出许多广泛运用的预条件子的原理和算法, 并在后面章节的数值实验分析大部分的预条件子.

3.1 块对角预条件子

块对角预条件子是由下列的块因子分解启发而来的.

$$\begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix} = \begin{pmatrix} I & 0 \\ BA^{-1} & I \end{pmatrix} \begin{pmatrix} A & 0 \\ 0 & S \end{pmatrix} \begin{pmatrix} I & A^{-1}B^T \\ 0 & I \end{pmatrix}. \quad (3.1)$$

这里 $S = -BA^{-1}B^T$ 便是所谓的 Schur 补.

预条件子 $P = \begin{pmatrix} A & 0 \\ 0 & -S \end{pmatrix}$, 作用在系数矩阵 F 上得到 $M = P^{-1}F = \begin{pmatrix} I & A^{-1}B^T \\ -S^{-1}B & 0 \end{pmatrix}$.

可以看到 $(M - I)\left(M + \frac{1+\sqrt{5}}{2}I\right)\left(M - \frac{1-\sqrt{5}}{2}I\right) = 0$. 这样 M 拥有三个特征值 $1, \frac{1+\sqrt{5}}{2}, \frac{1-\sqrt{5}}{2}$, 也就是说对于任何初始残量, M 拥有比较好的条件数, 运用 Krylov 子空间法时会在三步之内收敛^[16]. 但是我们通常不能直接用 A 和 S , 因为它们的稀疏性而求逆运算代价是巨大的, 并且存储是不易的. 所以我们需要近似它们, 也因此这样我们不可能三步内就得到收敛解, 如何得到好的近似也是当今的研究热点. 在本文试验中, 我们取 $\tilde{S} = -B \text{diag}(A)^{-1} B^T$ 作为 Schur 补的近似, 即在求 A^{-1} 时取 $D = \text{diag}(A)$ 作为近似, 如果 A 是主对角占优的话这样近似就更好了.

然后原问题就成求解下列问题:

$$\begin{pmatrix} A & 0 \\ 0 & \tilde{S} \end{pmatrix} \begin{pmatrix} z_u \\ z_p \end{pmatrix} = \begin{pmatrix} r_u \\ r_p \end{pmatrix}. \quad (3.2)$$

执行 $Pz = r$ 的算法 3.1 如下:

给定 $r = (r_u, r_p)^T$, 找到 $z = (z_u, z_p)^T$.

(1) 求解 $\tilde{S}z_p = r_p$;

(2) 求解 $Az_u = r_u$.

3.2 块上三角预条件子

同样由上式(3.1)的因式分解而来, 主要是把后面两项相乘. 预条件子 $P = \begin{pmatrix} A & B^T \\ 0 & S \end{pmatrix}$. 注意到 P 的第一行和 F 的第一行是一样的, 也就是给定的初始估计 $u_0 = 0$, 预条件方程 $Pz_0 = r_0$ 满足鞍点系统的第一个方程. 回忆等式(3.1), 我们可以看到 $P^{-1}F$ 的谱为 1, 并且预条件后的矩阵有最小的二范数, 因此使用 GMRES 方法会至多在两步内收敛^[16]. 在

这里我们同样取 $\tilde{S} = -B \text{diag}(A)^{-1} B^T$ 作为 Schur 补的近似. 这个预条件子花费比块对角预条件子稍微偏高, 除了要解带有近似的 \tilde{S} 还有乘上 B^T . 一般来说越好的近似收敛越快, 然而收敛却不易估计. 不过通过一些分析说明对于 Ossen 方程可以达到不依赖网格大小的收敛^[17].

$$\begin{pmatrix} A & B^T \\ 0 & \tilde{S} \end{pmatrix} \begin{pmatrix} z_u \\ z_p \end{pmatrix} = \begin{pmatrix} r_u \\ r_p \end{pmatrix}. \quad (3.3)$$

执行 $Pz = r$ 的算法 3.2 如下:

给定 $r = (r_u, r_p)^T$, 找到 $z = (z_u, z_p)^T$.

- (1) 求解 $\tilde{S}z_p = r_p$;
- (2) 更新 $r_u \leftarrow r_u - B^T z_p$;
- (3) 求解 $Az_u = r_u$.

3.3 块下三角预条件子

同样由上式 (3.1) 的因式分解而来, 主要是把前面两项相乘. 该预条件子和块上三角预条件子看起来比较类似, 但是由于速度项和压力项并不是对称的, 通常有限元等方法离散得到的速度项是比压力项复杂的, 所以跟块上三角预条件子还是不一样的.

$$\begin{pmatrix} A & 0 \\ B & \tilde{S} \end{pmatrix} \begin{pmatrix} z_u \\ z_p \end{pmatrix} = \begin{pmatrix} r_u \\ r_p \end{pmatrix}. \quad (3.4)$$

执行 $Pz = r$ 的算法 3.3 如下:

给定 $r = (r_u, r_p)^T$, 找到 $z = (z_u, z_p)^T$.

- (1) 求解 $Az_u = r_u$;
- (2) 更新 $r_p \leftarrow r_p - Bz_u$;
- (3) 求解 $\tilde{S}z_p = r_p$.

3.4 块 LU 预条件子

注意式 (3.1) 的因式分解. 把后两块相乘, 和块上三角预条件子类似, 只不过预条件子多乘了个矩阵.

$$\begin{pmatrix} I & 0 \\ BA^{-1} & I \end{pmatrix} \begin{pmatrix} A & B^T \\ 0 & \tilde{S} \end{pmatrix} \begin{pmatrix} z_u \\ z_p \end{pmatrix} = \begin{pmatrix} r_u \\ r_p \end{pmatrix}. \quad (3.5)$$

执行 $Pz = r$ 的算法 3.4 如下:

给定 $r = (r_u, r_p)^T$, 找到 $z = (z_u, z_p)^T$.

- (1) 求解 $\tilde{S}z_p = r_p - BD^{-1}r_u$;
- (2) 更新 $r_u \leftarrow r_u - B^T z_p$;
- (3) 求解 $Az_u = r_u$.

3.5 SIMPLE 预条件子

SIMPLE 预条件子是以前直接用作迭代算子的 SIMPLE 方法的一个变式, 之前用于有限体积法和有限差分法得到的离散格式. 这个方法通常基于以下步骤^[9]. 首先假设压力已经由前一次迭代所知, 然后速度便可以由动量方程求解得到. 因为初始压力只是一个猜测, 新得到的速度并不满足连续性方程. 因此便需要在下面的几个小步中速度与压力被校正以满足离散的连续性方程. 这个预条件子也是基于式 (3.1) 的分裂, 将式 (3.1) 的前两项组合得到的矩阵 $L = \begin{pmatrix} A & 0 \\ B & S \end{pmatrix}$, 该矩阵的一个近似就作为预条件子. 这个预条件子包含一个速度求解和压力项求解. 由于算法 3.5 每次迭代求出来的预条件子有一定精度, 该预条件子不能在以后的迭代中认为是不变的, 也因此该预条件子适合使用 GCR, 作为外层迭代. 可以证明 SIMPLE 预条件子可以提高预条件系统的全部谱. 一些特征值被聚在 1 附近, 还有一些则依赖于 Schur 补矩阵, 更多特征值分析可以参看[4].

执行 $Pz = r$ 的算法 3.5 如下:

给定 $r = (r_u, r_p)^T$, 找到 $z = (z_u, z_p)^T$.

- (1) 求解 $Az_u = r_u$;
- (2) 求解 $\tilde{S}z_p = r_p - Bz_u$;
- (3) 更新 $z_u \leftarrow z_u - D^{-1}B^T z_p$.

这里我们取 $D = \text{diag}(A)$, 但是仍然有其他选择, 仍需进一步的调查.

3.6 SIMPLER 预条件子

SIMPLER 预条件子是以前直接用作迭代算子的 SIMPLER 方法的一个变式, 它可以给出不依赖于 Reynolds 数的收敛. 和算法 3.5 相比则是求解压力项的不同, 用的是 $\tilde{S}z_p = r_p - BD^{-1}r_u$. 它包含两次压力求解和一次速度求解. 不过如果是该预条件子用于有限元离散后用 GCR 迭代法, 收敛效果很差甚至不收敛, 尤其是在细网格和内层系统精度低时^[18].

执行 $Pz = r$ 的算法 3.6 如下:

给定 $r = (r_u, r_p)^T$, 找到 $z = (z_u, z_p)^T$.

- (1) 求解 $\tilde{S}z_p = r_p - BD^{-1}r_u$;
- (2) 求解 $Az_u = r_u - B^T z_p$;
- (3) 求解 $\tilde{S}\delta z_p = r_p - Bz_u$;
- (4) 更新 $z_p \leftarrow z_p + \delta z_p$;
- (5) 更新 $z_u \leftarrow z_u - D^{-1}B^T \delta z_p$.

3.7 MSIMPLER 预条件子

MSIMPLER 是 Elman 等人提出的 SIMPLER 的一个改进^[4]. 考虑一个块因子分解, 它的 Schur 补是基于换位子逼近但是却建立在 SIMPLE 的近似的块因子分解. Elman 是观察非定常流的问题得到的矩阵 A_t 和时间有关的, $A_t = \frac{1}{\Delta t} Q_u$. 我们使用这种思想求解定常流的问题. MSIMPLER 的求解花费和 SIMPLER 是一样的, 因为 SIMPLER 就是取 $\hat{Q}_v = D$. 但是在求解 NS 方程的非线性迭代的一步, 不需要再次近似 Schur 补, 这是因为 Schur 补的逼近中的算子和非线性迭代部分的改变是无关的^[18]. 这和 SIMPLER 预条件子是相反的.

执行 $Pz = r$ 的算法 3.7 如下:

给定 $r = (r_u, r_p)^T$, 找到 $z = (z_u, z_p)^T$.

- (1) 求解 $\tilde{S}z_p = r_p - B\hat{Q}_v^{-1}r_u$;
- (2) 求解 $Az_u = r_u - B^Tz_p$;
- (3) 求解 $\tilde{S}\delta z_p = r_p - Bz_u$;
- (4) 更新 $z_p \leftarrow z_p + \delta z_p$;
- (5) 更新 $z_u \leftarrow z_u - \hat{Q}_v^{-1}B^T\delta z_p$.

3.8 UZAWA 预条件子

对于求解鞍点问题, UZAWA 迭代法是个不错的方法. 在参考文献[6, 19-21]中仔细讨论了线性 UZAWA 方法, 非线性 UZAWA 方法, 以及修正的非线性 UZAWA 方法. 而在本文讨论的 UZAWA 预条件子则是由基本的 UZAWA 迭代法得到的. 该预条件子有一个松弛因子 ω , 在这里我们取默认的 $\omega = 1.0$. 但如何选取更好的参数 ω , 仍需要更多的研究, 我们在后面的数值实验中通过比较迭代次数来找到比较好的 ω .

执行 $Pz = r$ 的算法 3.8 如下:

给定 $r = (r_u, r_p)^T$, 找到 $z = (z_u, z_p)^T$.

- (1) 求解 $Az_u = r_u$;
- (2) 更新 $z_p \leftarrow z_p + \omega(Bz_u - r_p)$.

3.9 投影预条件子

投影预条件子是由传统的求解 Navier-Stokes 方程的投影方法得到的. 在投影预条件子最后一步变式投影步骤也就是 $Bz_u = 0$, 这也就是说让 z_p 和 z_u 最后都是趋于稳定的值, 这就是该预条件子的思想.

执行 $Pz = r$ 的算法 3.9 如下:

给定 $r = (r_u, r_p)^T$, 找到 $z = (z_u, z_p)^T$.

- (1) 求解 $Az_u = r_u$;
- (2) 更新 $z_p \leftarrow z_p + Bz_u$;
- (3) 计算 $z_u \leftarrow z_u - B^T(BB^T)^{-1}Bz_u$.

3.10 DGS 预条件子

DGS 预条件子是由解决 Navier-Stokes 方程的多重网格法的一个磨光算子得到的, 也就是经典分布式 Gauss-Seidel 磨光算子. 在 DGS 预条件子中, $A_p = BB^T$.

执行 $Pz = r$ 的算法 3.10 如下:

给定 $r = (r_u, r_p)^T$, 找到 $z = (z_u, z_p)^T$.

- (1) 求解 $Az_u = r_u$;
- (2) 求解 $A_p \delta z_p \leftarrow r_p - Bz_u$;
- (3) 计算 $z_u \leftarrow z_u + B\delta z_p$;
- (4) 计算 $z_p \leftarrow -BB^T \delta z_p$.

3.11 LSC 预条件子

LSC 预条件子也就是所谓的最小二乘换位子预条件子. 这个方法很适合标准有限元方法, 但是只用于稳定的有限元. 即使是稳定化的有限元也是不行的, 因为它需要顾及两个调节参数, 因此我们只是用稳定化的有限元^[4]. A_p 是压力空间离散的对流扩散算子, M_u 指的是速度质量矩阵, M_p 则是压力质量矩阵. 思想就是逼近矩阵算子 A_p , 使得换位子 $\varepsilon = L\nabla - \nabla Q_k$ 变小, 其中 $L = -\nu \Delta + u_k \cdot \nabla$, 这是通过最小二乘问题做到的. 对于矩阵 A_p 的第 j 列, 最小问题有着以下形式:

$$\min \| (M_u^{-1} A M_u^{-1} B^T)_j - M_u^{-1} B^T M_p^{-1} (A_p)_j \|_{M_u}. \quad (3.6)$$

执行 $Pz = r$ 的算法 3.11 如下:

给定 $r = (r_u, r_p)^T$, 找到 $z = (z_u, z_p)^T$.

- (1) 求解 $Az_u = r_u$;
- (2) 求解 $A_p \delta z_p \leftarrow r_p - Bz_u$;
- (3) 计算 $z_u \leftarrow z_u + B\delta z_p$;
- (4) 求解 $A_p z_p \leftarrow -BAB^T \delta z_p$.

在 LSC 预条件子中, 我们取 $A_p = BB^T$.

3.12 PMM 预条件子

Schur 补, 也就是 $-BA^{-1}B^T$, 可以看做是 $-\text{div} \cdot \frac{1}{\nu} \Delta^{-1} \text{div}$ 的近似. 如果我们交换算子, 那么 $\text{div} \cdot \text{div} = \Delta$, 然后又由于有限元方法, 我们可以得到压力质量矩阵 M_p , 它的每个元素是

$$m_{ij} = \int \frac{1}{\nu} \psi_i \psi_j d\Omega. \quad (3.7)$$

这里 ψ_i 是对于压力项的基函数. 算法的每一步都可以在低于外层 GCR 迭代法精度下执行. 并且该算法相比块上三角预条件子相比求解压力项代价更小^[4].

完整的 PMM 预条件子可以写作

$$P = \begin{pmatrix} A & B^T \\ 0 & -M_p \end{pmatrix}. \quad (3.8)$$

执行 $Pz = r$ 的算法 3.12 如下:

给定 $r = (r_u, r_p)^T$, 找到 $z = (z_u, z_p)^T$.

- (1) 求解 $M_p z_p = r_p$;
- (2) 更新 $r_u \leftarrow r_u - B^T z_p$;
- (3) 求解 $A z_u = r_u$.

3.13 AL 预条件子

AL(增广拉格朗日)预条件子是 Benzi 为稳定有限元方法提出的, 后来也可用于稳定化的有限元. 这个方法主要在速度矩阵块增加一个惩罚项 $\gamma B^T W^{-1} B$, 这里的 γ 比较小, W 是缩放矩阵, 为了保持矩阵 A_γ 的稀疏性 W 通常是压力质量矩阵的对角矩阵^[22].

方程 (1.9) 被替代于

$$\begin{pmatrix} A_\gamma & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} f_\gamma \\ g \end{pmatrix}. \quad (3.9)$$

这里 $A_\gamma = A + \gamma B^T W^{-1} B$, $f_\gamma = f + \gamma B^T W^{-1} g$. 我们由连续性方程很容易证明式 (3.9) 是等价于式 (1.9) 的.

理想的预条件子 $P = \begin{pmatrix} A_\gamma & B^T \\ 0 & S \end{pmatrix}$. 这里 $S^{-1} = -\nu M_p - \gamma W^{-1}$, M_p 表示近似的压力质量矩阵, 通常情况下 W 和 M_p 是一样的.

执行 $Pz = r$ 的算法 3.13 如下:

给定 $r = (r_u, r_p)^T$, 找到 $z = (z_u, z_p)^T$.

- (1) 求解 $S z_p = r_p$, 也就是 $z_p = (-\nu M_p - \gamma W^{-1}) r_p$;
- (2) 更新 $r_u \leftarrow r_u - B^T z_p$;

(3)求解 $A_\gamma z_u = r_u$.

参数 γ 的选取是很关键的. 取值比较大的参数 γ 会降低外层迭代的数量, 但是会使得内层方程更难解, 但如果参数 γ 太小的话, 外层迭代的数目就会增加. 一般情况下我们会取 $\gamma = 1$, 但要取得更优的值, 还要根据粘度 ν 来调整它的值. 尽管 AL 的收敛和雷诺数和网格情况无关, AL 还是有缺点的. 首先 A_γ 包含交叉导数, 因此如果使用 Stokes 方程的简化形式的话通常会得到两倍于 (2D) 或者三倍 (3D) 于 Picard 迭代出来的原始矩阵, 但大部分的有限元方法使用的都不是简化形式. 但是如果使用的是 Taylor-Hood 有限元而不是 Crouzeix-Raviart 元, A_γ 的大小聚会比不是简化形式下的 A 还要大很多. 求解这个问题不是建立矩阵, 而是要用子矩阵进行矩阵和向量相乘. 这样的话计算就变得更昂贵. 还有一个问题就是如果粘度不是常数的话问题就更复杂了. 有数值实验表明在强变化的粘度下收敛速度会剧烈下降^[4, 22]. 如果方程 Reynolds 数较大的话, A_γ 就会变得非常不对称, 因此如果想收敛快一些的话 Benzi 发现使用多重网格法就能达到比较好的效果, 无论是几何多重网格法还是代数多重网格法.

4 数值实验

本次实验是求解二维定常流 Stokes 方程(2. 1), 满足 $\vec{f} = (0, 0)$, 区域 $\Omega = [0, 1] \times [0, 1]$, 满足 Dirichlet 边界: 在 $[0, 1] \times 1$ 上 $\vec{u} = (1, 0)$, 除此之外边界条件则是 $\vec{u} = (0, 0)$. 方程对应的速度流线见图 4. 1.

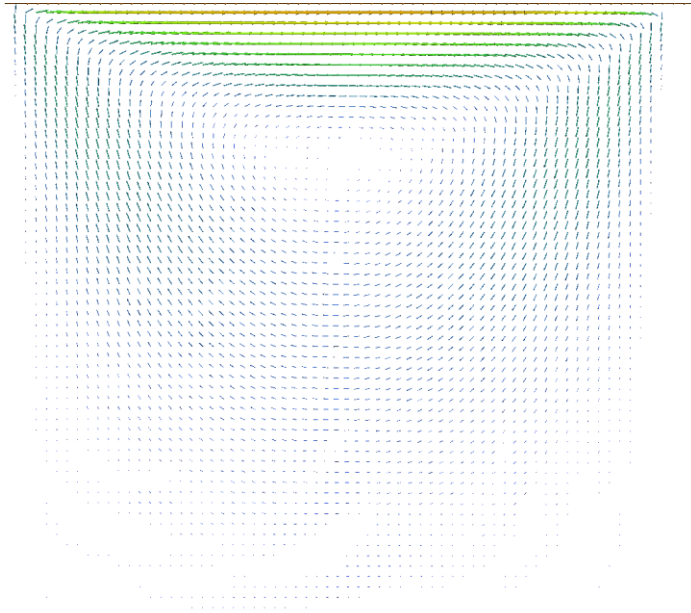


图 4.1 速度场

本次实验我们使用的是 FASP 求解器^[23], 以及 Fenics 软件, 实验在 Ubuntu16. 04 系统上进行. 我们用上面的其中的十个预条件子(块对角预条件子, 块上三角预条件子, 块下三角预条件子, 块 LU 预条件子, SIMPLE 预条件子, SIMPLER 预条件子, UZAWA 预条件子, 投影预条件子, DGS 预条件子, LSC 预条件子)和三种 Krylov 方法(MINRES, GMRES, FGMRES), 还有 AMG 方法, 进行了数据实验. 关于 AL 预条件子和 PMM 预条件子的表现状况, 可以参考文献^[2]. 关于 MSIMPLER 预条件子和其他 SIMPLE 型的对比表现状况, 可以参考文献^[4].

4.1 不同 Krylov 子空间方法配合各个预条件子的数值实验

4.11 实验说明

本次实验主要比较的是迭代次数, 设置最大迭代次数为 500, 如果 $\frac{\|r\|_2}{\|b\|_2} < 1e^{-6}$ 则停止迭代, 其中 b 指的是式(2.9)中的, r 是每次迭代的残量. 实验同时对速度块和压力块使用了经典 AMG 方法, 使用的是 Gauss-Seidel 磨光算子.

4.12 实验结果表格

我们测试了在 P2P1 元下不同 Krylov 子空间方法配合不同的预条件子, 结果如表 4.1.

表 4.1 P2P1 有限元下用不同的 Krylov 子空间方法各个预条件子的表现

| 迭代次数 | MINRES | GMRES | FGMRES |
|--------------|--------|-------|--------|
| 块对角预条件子 | >500 | 105 | 97 |
| 块上三角预条件子 | >500 | 49 | 48 |
| 块下三角预条件子 | >500 | 78 | 66 |
| 块 LU 预条件子 | >500 | 50 | 47 |
| SIMPLE 预条件子 | >500 | 71 | 64 |
| SIMPLER 预条件子 | >500 | 64 | 47 |
| UZAWA 预条件子 | >500 | 40 | 37 |
| 投影预条件子 | >500 | 78 | 74 |
| DGS 预条件子 | >500 | 78 | 74 |
| LSC 预条件子 | >500 | 57 | 44 |

4.13 实验结论

横向比较表 4.1, 我们可以发现 MINRES 方法表现的不好, 不仅收敛的很慢, 而且有的甚至无法收敛, GMRES 方法和 FGMRES 表现虽相差不大, 但 FGMRES 方法还是表现的更好, 可见后者却是均比前者改进不少.

纵向比较表 4.1, 我们可以发现 UZAWA 预条件子迭代次数是最少的, 表现的是最好的. 同样表现比较好的还有块上三角预条件子和块 LU 预条件子, 块上三角预条件子是比块下三角预条件子好了不少的. 但是不能因为一个问题的求解状况就能说明 UZAWA 预条件是好的, 因为影响结果的因素还有很多, 比如离散方法和网格等, 因此我们进行下面的实验.

4.2 不同的预条件子在不同网格下的数值实验

4.21 实验说明

而后我们再使用 FGMRES 方法, 分别比较这十种预条件子在 P_2-P_1 有限元和 P_2-P_0 有限元时的不同网格大小的表现状况, 结果见表 4.2 和 4.3.

4.22 实验结果表格

表 4.2 P_2P_0 有限元在不同网格大小下各个预条件子的表现

| 迭代次数 | $h = 1/8$ | $h = 1/16$ | $h = 1/32$ | $h = 1/64$ |
|--------------|-----------|------------|------------|------------|
| 块对角预条件子 | 43 | 61 | 91 | 133 |
| 块上三角预条件子 | 21 | 31 | 46 | 66 |
| 块下三角预条件子 | 26 | 40 | 63 | 99 |
| 块 LU 预条件子 | 20 | 30 | 45 | 66 |
| SIMPLE 预条件子 | 25 | 39 | 60 | 95 |
| SIMPLER 预条件子 | 19 | 30 | 48 | 76 |
| UZAWA 预条件子 | 13 | 14 | 13 | 13 |
| 投影预条件子 | 19 | 21 | 23 | 26 |
| DGS 预条件子 | 19 | 21 | 23 | 26 |
| LSC 预条件子 | 18 | 28 | 45 | 71 |

表 4.3 P_2P_1 有限元在不同网格大小下各个预条件子的表现

| 迭代次数 | $h = 1/8$ | $h = 1/16$ | $h = 1/32$ | $h = 1/64$ | $h = 1/128$ |
|--------------|-----------|------------|------------|------------|-------------|
| 块对角预条件子 | 43 | 65 | 97 | 141 | 203 |
| 块上三角预条件子 | 22 | 33 | 48 | 70 | 101 |
| 块下三角预条件子 | 27 | 43 | 66 | 106 | 166 |
| 块 LU 预条件子 | 21 | 32 | 47 | 70 | 101 |
| SIMPLE 预条件子 | 25 | 41 | 64 | 102 | 160 |
| SIMPLER 预条件子 | 19 | 30 | 47 | 75 | 117 |
| UZAWA 预条件子 | 45 | 39 | 37 | 37 | 36 |
| 投影预条件子 | 52 | 66 | 74 | 75 | 70 |
| DGS 预条件子 | 53 | 66 | 74 | 75 | 70 |
| LSC 预条件子 | 17 | 28 | 44 | 69 | 109 |

4.23 实验结论

通过观察表(4.2)和表(4.3),我们发现随着网格的细密,大部分预条件子所需要的迭代次数会变大.然而,UZAWA 预条件子,它的迭代次数却基本不会受网格尺寸的大小的影响,甚至出现了迭代次数递减的结果,可见其关于网格具有很好的鲁棒性.表现比较好的还有 DGS 预条件子和投影预条件子,它们受网格影响也比较小,并且迭代次数也极为一致.而通过比较相同预条件子不同有限元的 P_2-P_1 元和 P_2-P_0 元,可以发现不同的预条件子配合不同的离散方法,会得到不一样的迭代次数, P_2-P_0 有限元显得更具有优势,这也可能是跟选取的实验问题的求解区域有关系,在这里就不多分析.UZAWA 预条件子虽然在网格较粗的时候表现效果有的不如其他的,但是在网格更细的时候表现则是最好的,可见 UZAWA 预条件子确实是优秀的.

4.3 UZAWA 预条件子的最优参数选取数值实验

4.31 实验说明

我们知道, UZAWA 预条件子和一个参数 ω 相关,而 UZAWA 预条件子良好的表现说明有意义继续就该参数的选取进行理论研究,以期获得更快的收敛速度.上面的实验, UZAWA 预条件子部分我们所选取的参数均为 $\omega = 1.0$.然后我们在 P_2-P_0 有限元离散的 32×32 网格下,运用 FGMRES 方法,在不同的参数 ω 下比较收敛所需的迭代次数,来试图寻找比较好的参数 ω ,并且探讨它可能和什么有关,结果见表 4.4.

4.32 实验结果表格 (1)

表 4.4 在 P_2P_0 有限元的 32×32 网格下不同参数 ω 的表现状况

| | | | | | |
|----------|------|------|------|------|-------|
| ω | 0.1 | 0.2 | 0.3 | 0.6 | 1 |
| 迭代次数 | 40 | 39 | 38 | 37 | 37 |
| ω | 2 | 10 | 50 | 100 | 500 |
| 迭代次数 | 36 | 34 | 34 | 34 | 33 |
| ω | 1000 | 1500 | 2000 | 5000 | 10000 |
| 迭代次数 | 32 | 33 | 34 | 39 | 43 |

4.33 实验结论 (1)

观察上表可知,在这种情况下,在 $\omega = 1000$ 左右, UZAWA 预条件子的表现是最好的,并且达到最低的迭代次数的参数 ω 的取值范围都是比较大的.

4.34 实验结果表格 (2)

而后分别测试了 P2P0 有限元下不同网格的 UZAWA 预条件子的表现最好的参数 ω 大概值, 结果见表 4.5.

表 4.5 P2P0 有限元下不同网格的最好的参数 ω 的大概值

| 网格 | $h = 1/8$ | $h = 1/16$ | $h = 1/32$ | $h = 1/64$ | $h = 1/128$ |
|----------|-----------|------------|------------|------------|-------------|
| ω | 75 | 100 | 1000 | 4000 | 15000 |
| 迭代次数 | 36 | 34 | 32 | 32 | 31 |

然后我们还试了在参数 ω 取值为总网格节点数目的时候的收敛所需要的迭代次数, 结果见表 4.6. 不同网格大小对应着不同的网格单元数目, 而后离散所形成的对应的矩阵的规模(网格单元数目的平方)差别是很大的.

表 4.6 ω 取值等于不同节点数时的表现情况

| 节点数 | 81 | 289 | 1089 | 4225 | 16641 |
|------|----|-----|------|------|-------|
| 迭代次数 | 36 | 37 | 32 | 32 | 31 |

4.35 实验结论 (2)

在前面我们已经知道, UZAWA 预条件子受网格大小影响是很小的, 但是从表 4.5 可以看到最优参数 ω 的选取跟网格大小却是直接相关的, 并且影响很大. 同时我们发现随着网格单元的增大, 参数 ω 的值越大, 并且能取到不错的 ω 的范围也是大致增大的. 通过对比表 4.5 和表 4.6, 我们发现, 除了 $h = 1/16$ 的网格外, 参数 ω 的值与总网格单元的数目(即 h 的倒数的平方)或者说是节点的数目大致是一致的. 根据此实验, 虽然可以看到参数 ω 对 UZAWA 预条件子的影响其实并不是很大, 不过如果还是想用 UZAWA 预条件子得到更快的收敛, 我们就可以根据网格单元的数目来选取参数 ω , 从而达到优化数值的目的.

4.4 MG 预条件子的数值实验

4.41 实验说明

前文提到了 MG 方法本身也可以作为预条件子, 于是我们在这里就测试了 MG 作为预条件子的效果. 这里使用的是 LSCDGS 磨光算子, 测试了 P_2-P_1 元和 P_2-P_0 元下作为预条件子的 MG 方法的两种循环类型 V-循环和 W-循环的表现情况, 实验结果见表 4.7.

4.42 实验结果表格

表 4.7 不同网格下 MG 预条件子的表现情况

| 迭代次数 | $h = 1/8$ | $h = 1/16$ | $h = 1/32$ | $h = 1/64$ | $h = 1/128$ |
|------|-----------|------------|------------|------------|-------------|
| P2P0 | | | | | |
| V 循环 | 9 | 13 | 18 | 24 | 35 |
| W 循环 | 9 | 9 | 9 | 9 | 9 |
| P2P1 | | | | | |
| V 循环 | 14 | 21 | 23 | 25 | 26 |
| W 循环 | 14 | 20 | 23 | 23 | 23 |

4.43 实验结论

除了 P2P0 有限元比 P2P1 有限元在本次问题求解表现仍然较好外, 从表 4.7 我们可以看到 V-循环的迭代次数会随着网格细化增长迅速, 但是 W-循环则基本不变, 具有很好的鲁棒性, 可见 MG 作为预条件子效果也是不错的.

结 论

1 本文的主要贡献

本文大略介绍了 Navier-Stokes 方程, 有限元方法, Krylov 子空间方法和多重网格法, 针对 Stokes 方程离散后得到的鞍点问题, 将目前比较成熟的预条件子做了很详细的总结, 并用实验数据比较了大部分预条件子的表现.

通过实验我们首先注意到了不同种迭代法的迭代次数是不一样的, FGMRES 方法相比于 MINRES 和 GMRES 有着更好的表现效果. 然后我们发现了在不同的有限元空间里使用不同的 Krylov 子空间方法, UZAWA 预条件子的迭代次数是最小的, 预条件效果最好, 并且具有很好的鲁棒性. 而后本文再试图寻找 UZAWA 预条件子的最优参数 ω , 最后找到了参数 ω 的大致范围, 发现优秀的参数 ω 与网格单元的数目大致是一致的, 因此我们在使用 UZAWA 预条件子时可以根据网格单元的数目选取参数 ω 以得到更好的收敛效果. 最后我们发现 MG 方法作为预条件子也有不错的效果, W-型循环相比于 V-型循环鲁棒性更好.

2 下一步工作展望

我们在就 UZAWA 预条件子的最优参数 ω 的比较中, 由于比较好的参数 ω 的取值范围很大, 迭代次数大致差不多, 并没有非常精细的选取不同的参数 ω 来研究该参数究竟与什么相关, 因此这方面还仍待进一步的研究.

另外, 本文仍需要优化实验所采用的 FASP 代码, 寻找更好的求解线性方程组的软件包, 以更好更客观地比较各个预条件子的表现. 还有就是能否更好地利用该问题的块结构, 寻找可能存在的更好的预条件方法.

参 考 文 献

- [1] GALDI G P. An introduction to the Mathematical theory of the Navier-Stokes equations [M]. New York: Springer, 2011.
- [2] 胡泽均. 基于预处理 Navier-Stokes 方程的一类子系统[D]. 成都: 电子科技大学, 2013.
- [3] BENZI M. Preconditioning techniques for large liner systems: a survey [J]. Journal of Computational Physics, 2002, 182(2): 418-477.
- [4] SEGAL A, UR REHMAN M, VUIK C. Preconditioners for incompressible Navier-Stokes solvers [J]. Numerical Mathematics Theory Methods and Applications, 2010, 3(3): 245-275.
- [5] 谭宁波. 鞍点问题的预处理技术及其在 Navier-Stokes 方程中的应用[D]. 成都: 电子科技大学, 2013.
- [6] MIAO S. A new Uzawa-type method for saddle point problems [J]. Applied Mathematics and Computation, 2017, 300: 95-102.
- [7] DE NIET A C, WUBS F W. Two preconditioners for saddle point problems in fluid flows [J]. Int. J. Numer. Meth. Fluids, 2007, 54(4): 355-377.
- [8] NOTAY Y. An aggregation-based algebraic multigrid method [M]. Electronic transactions on numerical analysis, 2010, 37(6): 123-146.
- [9] KLAIJ C M, VUIK C. SIMPLE-type preconditioners for cell-centered, collocated finite volume discretization of incompressible Reynolds-averaged Navier-Stokes equations [J]. Int. J. Numer. Meth. Fluids, 2013, 17:830-849.
- [10] BOFFI D, BREZZI F, FORTIN M. Mixed Finite Elements, Compatibility Conditions, and Applications [M]. Springer-Verlag Berlin Heidelberg, 2008.
- [11] TAYLOR C, HOOD P. A numerical of the Navier-Stokes equations using the finite element technique [J]. Computers & Fluids, 1973, 1:73-100.
- [12] KONSHIN I, OLSHANSKII M, VASSILEVSKI Y. LU factorizations and ILU preconditioning for stabilized discretizations of incompressible Navier-Stokes equations [J]. Numerical Linear Algebra with Applications, 2017, 24(3), in press.
- [13] LIESEN J, STRAKOS Z. Krylov subspace methods: principles and analysis [M]. Oxford University, 2012.
- [14] 赵景余. 鞍点问题的数值解法[D]. 甘肃: 兰州大学, 2009.
- [15] 周福. 鞍点问题的预条件子研究[D]. 江西: 南昌大学, 2013.
- [16] BENZI M, GOLUB G H, LIESEN J. Numerical solution of saddle point problems [J]. Acta Numerica, 2005, 14:1-137.
- [17] KLAUWONN A, STARKE G. Block triangular preconditioners for nonsymmetric saddle point problems: field-of-values analysis [J]. SIAM Journal, 1999, 81(4):577-594.

- [18] LI C, SAGHIR A, BOERSTOEL G P. The Krylov accelerated SIMPLE(R) method for flow problems in industrial furnaces [J]. Int. J. Numer. Meth. Fluids, 2000, 33(7):1027-1040.
- [19] 叶海峰. 解鞍点问题的 UZAWA 算法及其收敛性分析[D]. 浙江: 浙江大学, 2005.
- [20] 庞宏奎, 黎稳. 求解对称鞍点问题的修正 UZAWA 方法[J]. 计算数学, 2009, 31(3): 269-278.
- [21] WANG N, LI J, LI G, et al. Variants of the Uzawa method for three-order block saddle point problem [J]. Applied Mathematics & Computation, 2017, 305: 118-202.
- [22] BENZI M, OLSHANSKII M A, An Augmented Lagrangian-Based Approach to the Ossen Problem [J]. SIAM J. sci. Comput. , 2006, 28(6): 2095-2113.
- [23] WU S, XU J, FENG C, et al. A multilevel preconditioner and its shared memory implementation for a new generation reservoir simulator [J]. 石油科学, 2014, 11(4): 540-549.

致 谢

本文正文到此就画上了句号，正如我的大学四年，也即将结束。在这里我想借此机会对许多人表示由衷的感谢。

首先，我要感谢我的导师张晨松老师和李崇君老师，他们在毕业论文上给了我莫大的帮助，指导和鼓励。在中科院的每周的组会上，以及私下的微信交流中，张晨松老师都能耐心细致的对我的问题提出建议和指导，给我提供了许多学习平台，让我长了不少见识，并且帮助我处理了程序上的许多疑问和错误。张老师亲和的气质和负责任的态度着实让我喜欢。李崇君老师在论文方面则促进了我对 UZAWA 预条件子的进一步探索，并且给予我的论文的撰写和修改提出了许多宝贵的意见，并且在四年的大学期间也教了我许多知识。

同时也要感谢许多的同学伙伴。同组会的范荣红向我讲述了许多混合有限元方面的知识，邹海峰则是告诉了我许多 Ubuntu 系统和 C++语言方面的知识，还有陈育炎学长带领我学习了 FASP 软件的大部分内容，徐雪枫则告诉了我许多矩阵分析和特征值的知识，这些都对我的论文起到了关键性的作用。同时感谢作为同学的刘景轩，岳江北，胡垚垚等人，他们在这一年生活中给我提供了许多帮助。

最后，感谢我的家人给予我大学四年的生活和精神上的支持，还有我的好朋友们，他们能一起快乐玩耍，也能在我灰心丧气的时候鼓励我，一起前进。