

王光辉, 谷湘潜, 公敬. 2012. 预处理 JFNK 方法及其在大气方程中的初步应用 [J]. 气候与环境研究, 17 (1): 13–19, doi:10.3878/j.issn.1006-9585.2011.09014. Wang Guanghui, Gu Xiangqian, Gong Jing. 2012. The preconditioning JFNK method and its initial application in weather equations [J]. Climatic and Environmental Research (in Chinese), 17 (1): 13–19.

预处理 JFNK 方法及其在大气方程中的初步应用

王光辉¹ 谷湘潜¹ 公敬²

¹ 中国气象科学研究院灾害天气国家重点实验室, 北京 100081

² 华北电力大学理学院, 北京 102206

摘 要 为提高数值求解大气方程的速度, 研究了预处理 JFNK (Jacobian-Free Newton-Krylov) 方法及其在大气方程中的应用。这是一种非线性外循环 Newton 迭代与线性内循环 Krylov 迭代相结合的快速算法, 其优点是进行外循环 Newton 迭代时不要求 Jacobian 矩阵的形成和存储; 它的有效性取决于内循环中线性系统的预处理。首先介绍了 JFNK 算法, 然后以浅水波方程为例, 描述了非线性残值的形成、预处理矩阵的构造及其在 JFNK 算法中的应用。试验结果表明: 对内循环线性系统进行适当的预处理, 能大幅度提高 JFNK 算法的运算速度。因而, JFNK 是一种值得在大气方程中推广应用的方法。

关键词 全隐式非线性格式 JFNK 算法 预处理方法 浅水波方程 Newton 迭代法。

文章编号 1006-9585 (2012) 01-0013-07 **中图分类号** P409 **文献标识码** A

doi:10.3878/j.issn.1006-9585.2011.09014

The Preconditioning JFNK Method and Its Initial Application in Weather Equations

WANG Guanghui¹, GU Xiangqian¹, and GONG Jing²

¹ State Key Laboratory of Severe Weather, Chinese Academy of Meteorological Sciences, Beijing 100081

² School of Mathematics and Physics, North China Electric Power University, Beijing 102206

Abstract The preconditioning JFNK (Jacobian-Free Newton-Krylov) method and its application in weather equations are studied to improve the computation speed in solving weather equations numerically. This method is based on the fully implicit discrete difference equations. It is a fast algorithm by the combination of the Newton iteration and the Krylov iteration. Its advantage is that the formation and storage of the Jacobian matrix in the outer Newton iteration is avoided. Its effectiveness depends on the preconditioning method to the Krylov inner loop. Firstly, the JFNK algorithm is introduced, and then an example about shallow water equations was given. The formation of nonlinear residuals, the construction of preconditioning matrix, and its application to JFNK are described in the example. It is shown that the computation speed of the JFNK algorithm can be greatly improved by a proper preconditioning for the linear system in the inner loop. Therefore, the algorithm is potential in the application of weather equations.

Key words fully implicit nonlinear scheme, JFNK algorithm, preconditioning method, shallow water equations, Newton iterative algorithm

收稿日期 2009-01-14 收到, 2011-10-25 收到修定稿

资助项目 国家自然科学基金 40875066、41075080、40745033

作者简介 王光辉, 男, 1962 出生, 研究员, 研究方向为数值天气预报。E-mail: wanggh@cma.cma.gov.cn

1 引言

大气圈是一个复杂多变的多尺度非线性系统 (Ji and Wang, 1995; Wang and Ji, 1995), 它的发展和演变受诸多因素 (如动力、热力和化学等) 的影响。综合考虑这些影响因素, 我们可以把大气体系描述成一组非常复杂的非线性非齐次偏微分方程组。到目前为止, 还没有人能从数学上求出这组方程的精确解。因此, 用数值分析的方法来求解大气方程组从而研究大气运动规律成为目前大气科学领域的主要的研究手段之一 (季仲贞, 1986; 季仲贞和王斌, 1991; 林万涛等, 2008)。

数值模拟大气方程最终往往会归结为求解一个大规模离散系统, 其中计算稳定性与计算速度是两个主要研究内容。对大规模离散系统的求解, 是许多科学与工程计算中数值模拟的中心问题, 而且通常在计算中是最耗时间的部分。在数值模拟过程中, 为了保障模式的计算稳定性和准确性 (Lilly, 1965; Gravel et al., 1993; Ramage, 1999), 所能选取的时间步长是较小的。针对不同的问题和不同的分辨率, 其时间步长可能是几分钟, 也可能是半小时, 至多取一小时, 其计算步数一般是较多的。此外, 为了保证计算的精度, 模式的空间分辨率又不能太低, 所以其总的计算量是巨大的。面对如此巨大的计算量, 在目前计算机资源远不能满足实际需要的情况下, 人们只能借助于数值算法来达到减少计算机存储和加快计算速度的目标, 而预处理迭代算法则是目前实现这个目标的重要手段。

预处理方法的优点体现在迭代次数与计算时间总量的减少。目前, 预处理主要归结为代数预处理 (Saad and Schultz, 1986; Chow, 2000; Chen, 2001; Benzi, 2002; 王光辉等, 2008) 和基于 PDES (Partial Differential EquationS) 的预处理 (Mousseau et al., 2000, 2002; Reisner et al., 2001, 2003)。前者对应的离散系统为线性系统。后者则对应非线性离散系统。代数预处理方法有简单、适应性广等特点, 但离散系统的线性化要求会使非线性方程的求解失去无可挽回的精度。基于 PDES 的预处理方法能处理非线性离散系统, 非线性离散系统往往更能反应原始方程,

但其求解往往会很复杂。首先必须用迭代法处理其非线性, 然后在每一时间步长内求解一个规模很大的线性方程组。应用预处理 JFNK (Jacobian - Free Newton - Krylov) 方法能有效地求解隐式非线性离散系统。这是一种对非线性系统进行双重循环求解的方法。外循环用 Newton 迭代方法将非线性离散系统在每一时间步内化为线性问题, 而且无须计算和存储 Jacobian 迭代矩阵。内循环用 Krylov 迭代法求解每一时间步内的线性系统。两重循环联合起来就称为 JFNK 方法。该方法的成功应用主要取决于对每一时间步内线性系统的预处理。预处理的方法有很多, 其最有效的是用原始方程的显式或半隐式等求解器作为预处理方法。有关这方面的研究取得了非常丰富的成果, 如: Brown and Saad (1990) 对相关 Krylov 子空间方法的基本原理和应用作了较为细致的研究; Mousseau et al. (2002) 为有效地求解隐式非线性差分方程, 并从理论上分析了预处理 JFNK 方法的计算精度与计算可行性; Knoll and Keyes (2004) 对预处理 JFNK 方法的应用和发展作了全面的总结; Mousseau et al. (2000) 用算子分裂法作为预处理方法对非平衡辐射扩散问题进行求解; Mousseau et al. (2002) 以二维浅水方程为应用背景, 提出了隐式非线性相容求解技巧, 该技巧的特点是用时间分裂求解法作为预处理方法, 通过对系统中最大时间尺度的有效消除, Krylov 迭代次数明显减少, 使 JFNK 方法的整体有效性增加; Reisner et al. (2003) 在求小尺度热驱动大气流的全隐式解时用半隐式离散预处理方法来减少 Krylov 迭代次数, 并将该预处理方法用到了隐式平衡飓风模型 JFNK 方法的求解中, 提出了所谓的隐式平衡方法 (Reisner et al., 2005)。为了量化新方法对比传统方法的有效性和准确性, 本文首次用降水湿泡模拟实验对隐式平衡求解器与半隐式方法进行了比较。降水湿泡试验显示: 对于给定的精度, 隐式平衡方法无论比一阶半隐式方法还是传统的蛙跳半隐式方法都更有效。此外, 线性多重网格方法、非线性多重网格方法及快速 Fourier 变换法等也被作为预处理方法应用于 JFNK 方法。

本文第二部分介绍预处理 JFNK 快速算法及基本步骤。第三部分以一维浅水波方程为例, 描

述预处理矩阵的构造及预处理方法在 JFNK 算法中的应用。构造浅水波模型的隐式非线性差分格式及其半隐式差分格式。比较了应用预处理 JFNK 方法求解隐式非线性差分格式及直接求解半隐式差分格式的结果。文章的最后我们对预处理 JFNK 方法作了简单的总结并对有关问题作了初步探讨。

2 预处理 JFNK 方法

2.1 预处理方法

JFNK 方法是一种嵌套式迭代算法。一般有 Newton 修正迭代外循环与建立 Krylov 子空间的内循环。在 Krylov 内循环内, 通常需要一个预处理器, 用以提高 JFNK 方法的有效性。预处理 JFNK 方法的目的是用来减少 GMRES (Generalized Minimal RESidual) 的迭代次数。GMRES 是一种基于 Arnoldi (或 Householder) 方法的非对称求解器 (Saad and Schultz, 1986), 构造近似解的基础是 Krylov 子空间, 而利用 Arnoldi (或 Householder) 方法求 Krylov 子空间的标准正交基是实现这一算法的关键。它的收敛其决于矩阵的特征值而不是矩阵的结构。对线性系统采用左乘或右乘一个预先选定的矩阵, 使其接近于单位矩阵, 然后再进行迭代求解, 这样能十分显著地减少求解的迭代次数, 从而达到减少计算量的目的。预处理器的选取方法有许多, 普通的方法是选择一个只需迭代几步的简单迭代方法作为预处理器。而一个有效的 JFNK 方法不但要回避 Jacobian 矩阵的形成, 而且其预处理器将比系统的 Jacobian 矩阵更为简单。最近发展起来用来预处理 JFNK 的预处理器是一种叫做基于物理意义 (或微分方程) 的预处理方法。这种方法非常有效, 其本质是基于描述物理问题的微分方程问题, 用一种简易离散格式的求解法去预处理较为复杂的离散格式。例如: 对于浅水波方程, 我们可以用半隐式格式、半隐式一半拉格朗日格式等的求解器或用离散格式的不完全 LU 分解法、时间分裂法等作为预处理器去求解非线性隐式方程。

总之, 对于非线性隐式格式的求解, 需把预处理、Jacobian-Free 和 GMRES 迭代法等几种有效工具结合起来才能达到在保证计算精度的前

提下提高其计算效率。

2.2 算法的基本步骤

2.2.1 Newton 迭代 (Dembo et al., 1982)

设隐式非线性差分方程为:

$$\mathbf{F}(\mathbf{x}) = 0, \quad (1)$$

$\mathbf{F}$ 是非线性残值向量; $\mathbf{x}$ 代表离散变量向量。用 Newton 迭代法求解方程 (1) 的公式为:

$$\mathbf{J}^k \delta \mathbf{x}^k = -\mathbf{F}(\mathbf{x}^k), \quad (2)$$

这里 $\mathbf{J}_{i,j}^k = \frac{\partial \mathbf{F}_i}{\partial \mathbf{x}_j}$, $\mathbf{x}^{k+1} = \mathbf{x}^k + \delta \mathbf{x}^k$, k 是迭代参数。

迭代终止标准

$$\frac{\|\mathbf{F}(\mathbf{x}^k)\|_2}{\|\mathbf{F}(\mathbf{x}^0)\|_2} < \epsilon, \quad (3)$$

这里取 $\epsilon = 1.0 \times 10^{-6}$ 。对于每一次迭代, (1) 为一线性方程组。其未知变量为 $n+1$ 时刻的量 (n 及其以前时刻的量已知)。

2.2.2 预处理

对线性方程 (1) 进行预处理以减少迭代求解时的迭代次数。设预处理阵为 $\mathbf{P}$, 那么经过预处理后方程 (1) 变为 (Mousseau et al., 2002):

$$\mathbf{J}^k \mathbf{P}^{-1} (\mathbf{P} \delta \mathbf{x}^k) = -\mathbf{F}(\mathbf{x}^k), \quad (4)$$

预处理器 $\mathbf{P}$ 的选取的要求是: $\mathbf{J}^k \mathbf{P}^{-1} \approx \mathbf{E}$ ($\mathbf{E}$ 为单位矩阵) 或 $\mathbf{J}^k \mathbf{P}^{-1}$ 的特征值聚集在 1 的附近。设 $\mathbf{P} \delta \mathbf{x}^k = \mathbf{v}$, 则预处理后的方程组可写成:

$$\mathbf{J}^k \mathbf{P}^{-1} \mathbf{v} = -\mathbf{F}(\mathbf{x}^k).$$

新的迭代矩阵是 $\mathbf{J}' = \mathbf{J}^k \mathbf{P}^{-1}$, 新的未知量为 $\mathbf{v} = \mathbf{P} \delta \mathbf{x}^k$, 那么方程 (4) 变为

$$\mathbf{J}' \mathbf{v} = -\mathbf{F}(\mathbf{x}^k). \quad (5)$$

2.2.3 求解方程

用重启 GMRES (m) (m 表示重启的次数) 迭代算法求解方程 (5) (Saad and Schultz, 1986)。

(1) 输入 $\mathbf{F}(\mathbf{x}^k)$ 及初始向量 $\mathbf{v}_0$, 最大迭代次数 m 及精度要求 ϵ 。这里 $\mathbf{v}_0 = \mathbf{P}^{-1} \mathbf{F}(\mathbf{x}^k)$, 在迭代过程中只用到 $\mathbf{J}'$ 与 $\mathbf{v}$ 的积, 所以无需计算 $\mathbf{J}'$ 。

(2) $\mathbf{r}_0 := -\mathbf{F}(\mathbf{x}^k) - \mathbf{J}' \mathbf{v}_0$, $\mathbf{q}_1 := \mathbf{r}_0 / \|\mathbf{r}_0\|_2$, $j=1$, 其中, $:=$ 表示定义、等于。

(3) $h_{i,j} := \mathbf{q}_i^T \mathbf{J}^k \mathbf{q}_j$, $i=1, 2, \dots, j$,

$$\mathbf{P}_{j+1} := \mathbf{J}^k \mathbf{q}_j - \sum_{i=1}^j h_{i,j} \mathbf{q}_i, \quad h_{j+1,j} := \|\mathbf{P}_{j+1}\|_2.$$

$\mathbf{q}_i^T$ 表示 $\mathbf{q}_i$ 的转置。

(4) 如果 $h_{j+1,j} < \epsilon$ 或 $j = m$, 则转下一步;

否则

$$\mathbf{q}_{j+1,j} = \mathbf{p}_{j+1}/h_{j+1,j}, j: = j+1,$$

转第 (3) 步。

(5) 求解最小二乘问题

$$\min_{z \in \mathcal{X}(\mathbf{J}^k, \mathbf{r}_0, j)} \|\mathbf{r}_0 - \mathbf{J}'\mathbf{z}\|_2 = \min\{\|\mathbf{r}_0 - \mathbf{J}'\mathbf{Q}_j\mathbf{y}\|_2, \mathbf{y} \in \mathbf{R}^j\},$$

其中, $\mathbf{R}$ 为实向量, 得 $\mathbf{y}_j, \mathbf{v}_j = \mathbf{v}_0 + \mathbf{Q}_j\mathbf{y}_j$, 这里 $\mathbf{Q}_j = [\mathbf{q}_1, \dots, \mathbf{q}_j]$ 。

(6) 如果 $\|\mathbf{J}'\mathbf{v}_j + \mathbf{F}(\mathbf{x}_j^k)\|_2 \leq \epsilon$, 则输出 $\mathbf{v}_j$, 结束; 否则 $\mathbf{v}_0 = \mathbf{v}_j$, 转第 (2) 步。

在上述 GMRES (m) 迭代求解方程 (5) 的过程中, 正交向量组 $\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_i$ 的计算与存储不能回避, 但可用重启动方法减少存储量。而对 Jacobian 的计算与存储可以回避。很明显, 在上面的 GMRES (m) 迭代中, 我们只用到 Jacobian 矩阵与一个向量的乘积 $\mathbf{J}'\mathbf{v}$, 对于小规模问题, 我们可以通过已知函数向量 $\mathbf{F}$ 。但由于 $\mathbf{F}$ 是一个满矩阵, 对于大规模问题, 若先求出 $\mathbf{J}'$, 然后再计算 $\mathbf{J}'\mathbf{v}$, 则需占用巨大的空间存储 $\mathbf{J}'$ 。为此我们用差商来逼近乘积 $\mathbf{J}'\mathbf{v}$, 即:

$$\mathbf{J}'\mathbf{v} \approx \frac{\mathbf{F}(\mathbf{x} + \tau\mathbf{v}) - \mathbf{F}(\mathbf{x})}{\tau}. \quad (6)$$

这个方法称为 Jacobian-Free, 这里 τ 是由下式确定的一个小参数,

$$\tau = \sum_{i=1}^m a x_i / (M \|\mathbf{v}_j\|_2),$$

其中, $a = 1.0 \times 10^{-8}$ (见 Mousseau et al., 2002)。

3 预处理 JFNK 算法在数值求解浅水波方程中的应用

3.1 浅水波模型

为了验证我们的预处理迭代方法的效果, 这里我们以浅水模型的隐式非线性格式为例, 用半隐式格式求解方法做预处理器进行预处理 JFNK 求解。并与半隐式格式的求解比较其计算有效性。我们选择无地形通量形式的一维浅水波方程进行数值实验。我们增加了一项附加力以保证解析解的封闭性。微分方程为 (Reisner et al., 2001):

$$\begin{cases} \frac{\partial h}{\partial t} + \frac{\partial uh}{\partial x} = f_h, \\ \frac{\partial uh}{\partial t} + \frac{\partial uuh}{\partial x} = -gh \frac{\partial h}{\partial x} + uf_h + hf_u, \end{cases} \quad (7)$$

这里

$$\begin{cases} f_h = (a_h\omega - a_u h_0 k) \sin(kx + \omega t) + a_h a_u k \sin 2(kx + \omega t), \\ f_u = (a_h g k - a_u \omega) \sin(kx + \omega t) - 0.5 a_u^2 k \sin 2(kx + \omega t), \end{cases} \quad (8)$$

其中, h 和 u 分别是流体的高度和速度, h_0 是初始波的平均高度, g 是重力加速度。 ω 是波的频率, k 是波数。

由文献 Reisner et al. (2001), 方程组 (1) 有下列解析解:

$$\begin{cases} u = a_u \cos(kx + \omega t), \\ h = h_0 - a_u \cos(kx + \omega t), \end{cases}$$

注意: 解析解被用来作为初始条件 (设 $t=0$)。在这个简单模型里我们还用到了周期边界条件。

下面对方程组 (7) 进行差分离散, 获得全隐式非线性离散和半隐式线性格式 (见 3.2 节和 3.3 节)。i 取 1, 2, 3..., N, 0, N+1 为边界点。

3.2 全隐式非线性格式

$$\begin{cases} \frac{h_i^{n+1} - h_i^n}{\Delta t} + \frac{(uh)_{i+1}^{n+1} - (uh)_{i-1}^{n+1}}{2\Delta x} = f_{h_i}^{n+1}, \\ \frac{(uh)_i^{n+1} - (uh)_i^{n+1}}{\Delta t} + \frac{(u^2 h)_{i+1}^{n+1} - (u^2 h)_{i-1}^{n+1}}{2\Delta x} = \\ -gh_i^{n+1} \frac{h_{i+1}^{n+1} - h_{i-1}^{n+1}}{2\Delta x} + u_i^{n+1} f_{h_i}^{n+1} + h_i^{n+1} f_{u_i}^{n+1}. \end{cases} \quad (9)$$

对应的非线性残值向量的分量为:

$$\begin{aligned} \mathbf{F}_{2i-1} &= h_i^{n+1} + \lambda (uh)_{i+1}^{n+1} - \lambda (uh)_{i-1}^{n+1} - h_i^n - \Delta t f_{h_i}^{n+1}, \\ \mathbf{F}_{2i} &= (uh)_i^{n+1} + \lambda \cdot \frac{(u^2 h^2)_{i+1}^{n+1}}{h_{i+1}^{n+1}} - \lambda \cdot \frac{(u^2 h^2)_{i-1}^{n+1}}{h_{i-1}^{n+1}} + \\ &\quad \lambda \cdot g \cdot h_k^{n+1} [h_{i+1}^{n+1} - h_{i-1}^{n+1}] - (uh)_i^n - \\ &\quad \Delta t \cdot \frac{(uh)_i^{n+1}}{h_{i+1}^{n+1}} f_{h_i}^{n+1} - \Delta t \cdot h_i^{n+1} f_{u_i}^{n+1}, \end{aligned}$$

其中, $\lambda = \Delta t / (2\Delta x)$ 。当 n 时刻的量为已知时, 则 $n+1$ 时刻的量就是牛顿迭代的未知量。

3.3 半隐式格式

$$\begin{cases} \frac{h_i^{n+1} - h_i^n}{\Delta t} + \frac{(uh)_{i+1}^{n+1} - (uh)_{i-1}^{n+1}}{2\Delta x} = f_{h_i}^{n+1}, \\ \frac{(uh)_i^{n+1} - (uh)_i^{n+1}}{\Delta t} + \frac{(u^2 h)_{i+1}^{n+1} - (u^2 h)_{i-1}^{n+1}}{2\Delta x} = \\ -gh_i^n \frac{h_{i+1}^{n+1} - h_{i-1}^{n+1}}{2\Delta x} + u_i^n f_{h_i}^{n+1} + h_i^n f_{u_i}^{n+1}. \end{cases} \quad (10)$$

3.4 改写半隐式方程

将半隐式方程写成 $\mathbf{P}\delta\mathbf{w} = -\mathbf{F}(\mathbf{w})$ 形式。设

$\delta h = h^{n+1} - h^n$, $\delta(uh) = (uh)^{n+1} - (uh)^n$. 则上述半隐式方程写为:

$$\begin{cases} \delta h_i + \lambda[\delta(uh)_{i+1} - \delta(uh)_{i-1}] = \Delta t f_{h_i}^{n+1} - \\ \lambda[(uh)_{i+1}^n - (uh)_{i-1}^n], \\ \delta(uh)_i + g\lambda h_i^n (\delta h_{i+1} - \delta h_i) = \\ \lambda[(u^2 h)_i^n - (u^2 h)_{i+1}^n] - g\lambda h_i^n [h_{i+1}^n - h_{i-1}^n] + \\ \Delta t f_{u_i}^{n+1} [u_i^n + h_i^n]. \end{cases} \quad (11)$$

很显然, 当方程组 (9) 中的 $n+1$ 时间层变量的值取为 n 层的值时, 对应的残值向量 $\mathbf{F}$ 正好等于方程组 (11) 右边的值组成的向量反号, 即: $-\mathbf{F}$. 于是, 最终能将方程组 (11) 写成 $\delta \mathbf{w} = -\mathbf{P}^{-1} \mathbf{F}(\mathbf{w})$ 的形式, 其中 $\delta \mathbf{w} = \{\delta h, \delta(uh)\}$. 由方程组 (11) 消去 $\delta(uh)$ 就得到关于 δh 的方程, 并求解, 最后在求出 $\delta(uh)$. 在 GMRES 迭代过程中, 初值向量 $\mathbf{v}_0$ 取 n 时刻变量的值. 对于 $\mathbf{P}^{-1}$, 我们并不需要显式求出, 因为在迭代过程中只用到 $\mathbf{P}^{-1}$ 与一个向量的乘积. 如果用任意已知向量 $\mathbf{v}$ 替代方程组 (11) 的右边, 则可得到 $\mathbf{P}^{-1}$ 与 $\mathbf{v}$ 的乘积.

差分方程组中的参数取值如下: $g=1.0 \text{ m} \cdot \text{s}^{-2}$, $h_0=1.0 \text{ m}$, $a_h=0.01 \text{ m}$, $a_u=0.01 \text{ m} \cdot \text{s}^{-1}$, $k=2 \text{ m}^{-1}$, $\omega=0.0 \text{ s}^{-1}$, $x \in [0, 2\pi]$, $t \in [0, 2\pi]$.

计算中取如下截断误差

$$T_E(N, M) = \left\{ \sum_{j=1}^N \sum_{i=1}^M [\phi(t_j, x_i) - \phi_i^j] \right\}^{1/2}$$

这里 $N=T/\Delta t$, $M=L/\Delta x$, $\phi(x, t)$ 表示 $u(x, t)$ 或 $h(x, t)$; ϕ_i^j 表示 $\phi(x_i, t_j)$ 在点 (x_i, t_j) 的离散解.

3.5 预处理结果比较

这部分我们将简要地说预处理前后求解 (9) 的结果进行比较. 先将半隐式差分格式 (10) 写成

$$\begin{cases} h_i^{n+1} + kS_{i+1}^{n+1} - kS_{i-1}^{n+1} = h_i^n + \Delta t f_{h_i}^{n+1}, \\ S_i^{n+1} + gh_i^n h_{i+1}^{n+1} - gh_i^{n+1} kh_{i-1}^{n+1} = d_i^n, \end{cases} \quad (12)$$

这里 $S_i^n = (uh)_i^n$, $d_i^n = -k(u_{i+1}^n P_{i+1}^n - u_{i-1}^n P_{i-1}^n) + \Delta t u_i^n f_{h_i}^{n+1}$, $\lambda = \Delta t / (2\Delta x)$. 我们将先通过求解 $h(x, t)$ 和中间变量 S , 然后由 $S=uh$ 求 $u(x, t)$.

首先我们通过方程组 (12) 求解方程组 (10), GMRES (m) 迭代收敛误差取为 $\epsilon < 10^{-6}$, 重启参数 $m=10$. 然后用半隐式格式 (10) 的求解器做预处理器去求解全隐式非线性格式 (9),

外循环 Newton 迭代收敛误差与内循环 GMRES (m) 迭代收敛误差均取为 $\epsilon < 10^{-6}$.

为测试预处理前后 GMRES 迭代次数、运算时间与时空分辨率及与收敛精度的关系, 我们通过: 固定时间步数, 空间分辨率逐步提高; 固定空间步数, 时间分辨率逐步提高及时空步长按一定比率同步提高等 3 个试验比较预处理效果.

表 1 为当时间步数固定为 $N=320$ 时, 随着空间分辨率的提高 (M 增大), 预处理前后, 求解平均每时间步长 GMRES 迭代次数和总的 CPU 时间的比较. 从表中的数字看出, 随着空间分辨率的提高, 预处理前后每时间步的平均迭代次数逐步增加, 但预处理后的迭代次数及总的 CPU 时间减少幅度较大. 而且, 从变趋势看, 随着计算规模增大, 预处理后的平均迭代次数及总的计算时间减少的比率越大. 如: 当 $M=50$, 预处理后的平均迭代次数虽然只有预处理前的 40%, 但总的计算时间却相当. 这是因为预处理所花的时间基本抵消了线性系统迭代次数减少所节省的时间. 而当空间步数 800 时, 其迭代次数为预处理前求解迭代次数的 23%. 所用总的 CPU 时间仅为预处理前的一半多一点. 还有, 由于我们在每次牛顿迭代和 GMRES 迭代所用的初值都是前一时间步的数值解, 所以, 不用预处理求解线性系统所需的迭代次数就很少, 尽管如此, 预处理效果仍然很明显.

表 1 时间步数固定为 $N=320$, 预处理前后平均每时间步长 Krylov 迭代次数和总的 CPU 时间的比较

Table 1 Comparison of averaged Krylov iterative number per time step and the total CPU time with and without precondition when the total number of the time steps is fixed as $N=320$

M	预处理前		预处理后	
	平均迭代次数	总 CPU 时间/s	平均迭代次数	总 CPU 时间/s
50	5	0.891000	2	0.875000
100	5	1.188000	2	1.172000
200	5	2.859000	2	2.406000
400	6	4.375000	2	3.000000
800	13	13.93800	3	7.578000

表 2 为当空间步数固定为 $M=400$ 时, 随着时间分辨率的提高 (N 增大), 预处理前后平均每

时间步长 GMRES 迭代次数和总的 CPU 时间的比较。随着时间步长减小, 预处理前后的平均迭代次数均呈现减少的趋势, 这是因为时空步长的比在逐渐减小, 迭代矩阵的条件数变好, 故而有迭代次数由减少的变化。虽然如此, 预处理后的效果仍然很明显。

表 2 空间步数固定为 $M=400$, 预处理前后平均每时间步 Krylov 迭代次数和总的 CPU 时间的比较

Table 2 Comparison of averaged Krylov iterative number per spacial step and the total CPU time with and without precondition when the total number of the time steps is fixed as $M=400$

N	预处理前		预处理后	
	平均迭代次数	总 CPU 时间/s	平均迭代次数	总 CPU 时间/s
100	36	4.453000	3	1.125000
160	15	3.531000	2	1.625000
220	9	3.594000	2	2.266000
280	7	3.985000	2	2.891000
300	7	4.219000	2	3.234000

为了测试我们所采用的预处理迭代法是优化稳定的, 我们用时空同步加密的办法计算了预处理前后总的 GMRES 迭代次数和求解所需总的 CPU 时间。从表 3 看出, 平均每时间步的迭代次数随系数矩阵的规模增大而减少, 预处理效果仍非常明显。

表 3 当时间步长和空间步长以一定的比率减少 (即 τ 保持不变) 时, 预处理前后平均每时间步 Krylov 迭代次数和总的 CPU 时间的比较

Table 3 Comparison of averaged Krylov iterative number per time step and the total CPU time with and without precondition when both the time step length and the spacial step length are reduced according as a certain ratio (i. e. τ keeps a constant)

时空步长按 比率变化 ($M \times N$)	预处理前		预处理后	
	平均迭代次数	总 CPU 时间/s	平均迭代次数	总 CPU 时间/s
50×20	30	0.094000	3	0.047000
100×40	25	0.313000	3	0.125000
200×80	19	1.188000	3	0.563000
400×160	15	3.547000	3	1.641000
800×320	13	13.969000	3	7.609000

图 1 表示时空网格剖分因子 (即: 时空网格被等分成 2^κ 份, κ 即为剖分因子) 与近似解截断

误差间的关系。反映了方程组 (9) 的数值解随着时空网格的加密将趋于方程组 (7) 的解。

图 2 表示迭代收敛误差的对数与平均每时间步长迭代次数的关系。图形反应了预处理 JFNK 方法更快的收敛性。图 2 中带有星号的连线是预处理前的结果, 从图形的走向我们可看出: 随着迭代收敛误差的减少, 平均每时间步长的 Krylov 迭代次数迅速增加。而经过预处理后 (见图 2 中带有空心圆的连线), 平均每时间步长的 Krylov 迭代次数的增加则相对缓慢。

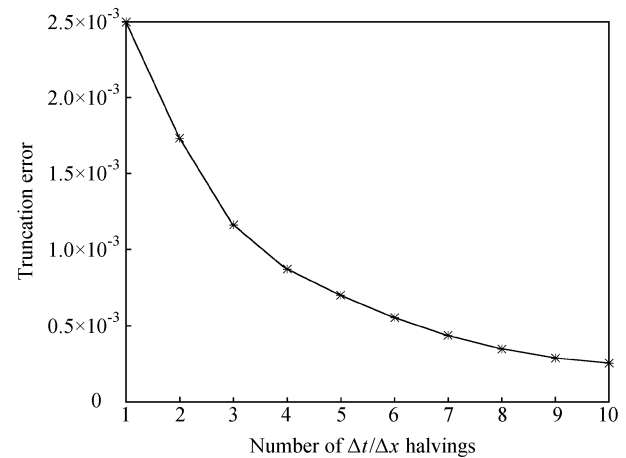


图 1 剖分因子与截断误差的关系

Fig. 1 The relation between the partition factor and the truncation error

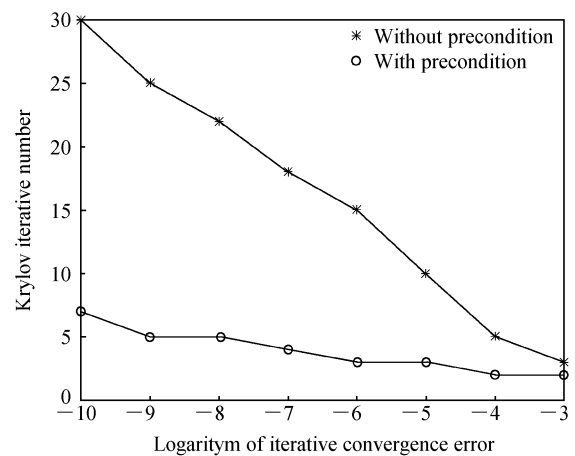


图 2 $M=400$, $N=160$ 时, 迭代收敛误差的对数与平均每时间步迭代次数的关系

Fig. 2 The relation between the logarithm of the iterative convergence error and the averaged iterative number per time step when $M=400$, $N=160$

4 总结和讨论

本文以一维浅水方程为例对隐式非线性差分格式的求解作了一些初步尝试, 对预处理 JFNK 快速算法从理论结果、具体构造及数值验证做了较为全面的分析。预处理 JFNK 快速算法虽然在实际应用中需用到双重迭代, 但如果预处理方法及相关参数选取合适, 将会带来非常高的计算效率。特别是超大规模的计算, 这种效果会更明显。本文仅选取为半隐式差分格式的求解器做预处理方法, 进行了数值试验。实际上类似的预处理方法有很多。从理论上讲, 用连续方程构造的差分格式的求解器做预处理器比代数预处理方法构造的预处理器更为有效。所以是一种值得在数值天气预报中推广应用的快速算法。

参考文献 (References)

- Benzi M. 2002. Preconditioning techniques for large linear system; a survey [J]. *J. Comput. Phys.*, 182: 418–477.
- Brown P N, Saad Y. 1990. Hybrid krylov methods for nonlinear systems of equations [J]. *SIAM J. Sci. Stat. Comput.*, 11 (3): 450–481.
- Chen K. 2001. An analysis of sparse approximate inverse preconditioners for boundary integral equations [J]. *SIAM J. Matrix Anal. Appl.*, 22 (4): 1058–1079.
- Chow E. 2000. A priori sparsity patterns for parallel sparse approximate inverse preconditioners [J]. *SIAM J. Sci. Comput.*, 21: 1804–1822.
- Dembo R S, Eisenstat S C, Steihaug T. 1982. Inexact Newton methods [J]. *SIAM Numer. Anal.*, 19 (2): 400–408.
- Gravel S, Staniforth A, Côté J. 1993. A stability analysis of a family of baroclinic semi-Lagrangian forecast models [J]. *Mon. Wea. Rev.*, 121: 815–824.
- 季仲贞. 1986. 计算地球物理流体力学中的非线性不稳定问题 (续) [J]. *力学进展*, 16 (4): 305–318. Ji Zhongzhen 1986. On the nonlinear computational instability in computational geophysical fluid dynamics [J]. *Advances in Mechanics (in Chinese)*, 16 (4): 305–318.
- 季仲贞, 王斌. 1991. 再论发展方程差分格式的构造和应用 [J]. *大气科学*, 15 (2): 1–10. Ji Zhongzhen, Wang Bin. 1991. Further discussion on the construction and application of difference scheme of evolution equations [J]. *Chinese Journal of Atmospheric Sciences (in Chinese)*, 15 (2): 1–10.
- Ji Zhongzhen, Wang Bin. 1995. Some splitting methods for equations of geophysical fluid dynamics [J]. *Advances in Atmospheric Sciences*, 12 (1): 109–113, doi: 10.1007/BF02661293.
- Knoll D A, Keyes D E. 2004. Jacobian-free Newton–Krylov methods: a survey of approaches and applications [J]. *J. Comput. Phys.*, 193: 357–397.
- Lilly D K. 1965. On the computational stability of numerical solutions of time-dependent nonlinear geophysical fluid dynamics problems [J]. *Mon. Wea. Rev.*, 93 (1): 11–25.
- 林万涛, 张博, 林一骅. 2008. 简单海气耦合模式大气运动方程的非线性对 ENSO 循环模拟的影响 [J]. *气候与环境研究*, 13 (3): 253–259. Lin Wantao, Zhang Bo, Lin Yihua. 2008. Influences of nonlinear factors of governing equations for the atmosphere on ENSO Cycles in a Coupled Ocean–Atmosphere Model. *Climatic and Environmental Research (in Chinese)*, 13 (3): 253–259.
- Mousseau V A, Knoll D A, Rider W J. 2000. Physics-based preconditioning and the Newton–Krylov method for non-equilibrium radiation diffusion [J]. *J. Comput. Phys.*, 160: 743–785.
- Mousseau V A, Knoll D A, Reisner J M. 2002. An implicit nonlinearly consistent method for the two-dimensional shallow-water equations with Coriolis force [J]. *Mon. Wea. Rev.*, 130: 2611–2625.
- Ramage A. 1999. A multigrid preconditioner for stabilised discretisations of advection-diffusion problems [J]. *Journal of Computational and Applied Mathematics*, 110: 187–203.
- Reisner J, Mousseau V, Knoll D. 2001. Application of the Newton–Krylov method to geophysical flows [J]. *Mon. Wea. Rev.*, 2001, 129: 2404–2415.
- Reisner J, Wyszogrodzki A, Mousseau V, et al. 2003. An efficient physics-based preconditioner for the fully implicit solution of small-scale thermally driven atmospheric flows [J]. *J. Comput. Phys.*, 189: 30–44.
- Reisner J M, Mousseau V A, Wyszogrodzki A A, et al. 2005. An implicitly balanced hurricane model with physics-based preconditioning [J]. *Mon. Wea. Rev.*, 133: 1003–1022.
- Saad Y, Schultz M H. 1986. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems [J]. *SIAM J. Sci. Stat. Comput.*, 7: 856–869.
- Wang Bin, Ji Zhongzhen. 1995. A class of explicit forward time-difference square conservative Scheme [J]. *Acta Mechanica Sinica*, 11 (1): 8–14.
- 王光辉, 陈峰峰, 沈学顺. 2008. 一种快速预处理算法及其在浅水波方程中的应用 [J]. *高原气象*, 27 (5): 978–985. Wang Guanghui, Chen Fengfeng, Shen Xueshun. 2008. A fast preconditioned algorithm and its application to shallow water equations [J]. *Plateau Meteorology (in Chinese)*, 27 (5): 978–985.