求解偏微分方程的时间滤波器方法*

张亚兰，黄鹏展†

(新疆大学 数学与系统科学学院，新疆 乌鲁木齐 830017)

摘　要： 偏微分方程在计算流体力学领域有着广泛的应用, 但大多数的偏微分方程无法直接解得显式解, 所以建立一种能高效求解的数值方法是至关重要的. 对于数值计算方法来说, 精度和有效性是一个算法的核心. 时间滤波算法是一种基于复杂系统原始计算代码的数值后处理算法. 该方法是在原始数值格式的基础上添加一个简单的时间滤波器来提高时间精度, 而不额外增加计算复杂度, 因而该方法在流体问题数值求解中得到了广泛运用. 梳理了一些基于时间滤波器的数值方法以及时间滤波器在几种方程中的应用. 最后, 基于时间滤波器, 针对 Navier-Stokes 方程, 提出变时间滤波器.

关键词： 偏微分方程; 时间精度; 时间滤波器; 有限元方法

DOI： 10.13568/j.cnki.651094.651316.2024.01.14.0002

中图分类号： O241.82　**文献标识码：** A　**文章编号：** 2096-7675(2025)01-0036-012

引文格式： 张亚兰，黄鹏展. 求解偏微分方程的时间滤波器方法[J]. 新疆大学学报(自然科学版中英文)，2025，42(1)：36-47.

英文引文格式： ZHANG Yalan，HUANG Pengzhan. Time filter method for solving partial differential equations[J]. Journal of Xinjiang University(Natural Science Edition in Chinese and English)，2025，42(1)：36-47.

Time Filter Method for Solving Partial Differential Equations

ZHANG Yalan, HUANG Pengzhan

(*School of Mathematics and System Sciences, Xinjiang University, Urumqi Xinjiang 830017, China*)

Abstract： Partial differential equations have a wide range of applications in many areas of computational fluid dynamics. However, most of these equations cannot be solved directly. Hence it is crucial to establish numerical methods that can solve them efficiently. For numerical methods, accuracy and effectiveness are the core of an algorithm. The filtering algorithm is a numerical post-processing algorithm based on the original computational code of a complex system. The method is based on adding a simple time filter to the original numerical scheme to improve the time accuracy without additional computational complexity, and is widely used in fluid problems. In this paper, we describe the time filters, some numerical methods based on time filter, and the application of time filters to several equations. Moreover, we design the time variable filter to the Navier-Stokes equations.

Key words： partial differential equations; time accuracy; time filter; finite element method

0　引言

几十年来, 时间滤波器作为提高精度和控制离散化技术引入非物理模式影响的计算工具一直被探索. 滤波器最早是由 Robert 在 1966 年提出的一种频率滤波器[1], 最初应用于使用气象模型进行天气预报的蛙跳格式. Robert 后来在数值天气预报的国际会议上介绍了这种滤波器的半隐式时间积分算法[2]. 1972 年, Asselin 通过分析发现该滤波器消除了蛙跳格式中所产生的数值振荡, 并阻尼了高频波[3], 被称为 Robert-Asselin (RA) 滤波器, 这是第一个时间滤波器. 为了提高这种滤波蛙跳格式的振幅精度, Williams 等利用高阶时间导数近似法对滤波器的振幅精度进行了多次改进[4−6], 最终形成了所谓的 RAW 滤波器. 最近的其它研究中, Li 和 Trenchea 推导出了高阶 RA (hoRA) 时间滤波器, 提高了蛙跳格式的稳定性和精度[7]. Guzel 和 Trenchea 证明了当 hoRA 格式与 Williams 步骤相结合时, 精度和稳定性都得到进一步提高[8].

* **收稿日期：** 2024-01-14
基金项目： 国家自然科学基金“双区域自然对流耦合模型的高效数值方法研究”（12361077）.
作者简介： 张亚兰（1998—），女，硕士生，从事偏微分方程数值解法的研究.
† **通讯作者：** 黄鹏展（1983—），男，博士，教授，主要从事偏微分方程数值解法的研究，E-mail：hpzh@xju.edu.cn.

早期对滤波格式的分析主要集中在常微分方程的处理上, 时间滤波器在偏微分方程离散化中的应用目前仍在探索中, 我们可以找到一些早期的相关工作[9−11]. 近年来, 时间滤波器应用于流体方程来控制非物理振荡[12−13]. 2018 年, Guzel 等通过在完全隐式或向后欧拉方法中添加简单的时间滤波器, 提高了时间精度[14]. 这种方法是模块化的, 只需要添加几行额外的代码. 关于误差估计和变时间步长算法是容易实现的, 同时作者在带有曲率减少时间滤波器的向后欧拉方法中引入了等价的两步二阶 A 稳定的线性方法. 研究指出, 时间滤波器是一种能够提高时间精度的工具. 紧接着, 基于 Guzel 等的工作, 许多计算流体动力学模型[15−21]采用了时间滤波器来提高时间精度, 包括 Stokes/Darcy 模型、Magnetohydrodynamics 模型、Navier-Stokes (NS) 模型、地热模型、双扩散自然对流模型、线性双曲模型和 NS/NS 耦合模型等. 另外, 研究者们将常数时间步扩展到变时间步长中, 获得了变时间步长滤波器, 从而在时间滤波器基础上提出了变时间步长的自适应算法[22−25].

1　几种时间滤波器

本节中, 介绍了几种常见的时间滤波器, 包括 RA 滤波器、RAW 滤波器和高阶的 RA 滤波器. 通过以下方程, 我们来阐述这几类滤波器的构造和发展.

首先, Asselin 根据如下方程分析了频率滤波器[3], 得到了第一个时间滤波器 (RA 滤波器). 所考虑方程为

$$u(t)=e^{i\omega t},$$

其中: ω 为角频率, $i=\sqrt{-1}$. 滤波函数为

$$\overline{u(t)}=u(t)+0.5\tau[\overline{u(t-1)}-2u(t)+u(t+1)],$$

其中: τ 是滤波器参数, 令 $\overline{u(t)}=\overline{X}u(t)$, $\overline{X}=\big((2-\tau)^2+2\tau^2(1-\cos\omega)e^{i\omega}\big)/\big((2-\tau)^2+4\tau(1-\cos\omega)\big)=R(\tau,\omega)e^{i\delta(\tau,\omega)}$, R 为振幅响应, δ 为相移. 进而对具有蛙跳、隐式和半隐式的滤波器进行了线性分析.

其次, 根据下面的复振荡方程[7,26]得到 RA, RAW 和 hoRA 时间滤波器

$$\frac{\mathrm{d}u}{\mathrm{d}t}=i\omega u \tag{1}$$

其中: ω 是一个实常数, $i=\sqrt{-1}$. 取定时间步长 Δt, 则在时间 $t_n=n\Delta t$ 处的精确解 u 表示为 $u(t_n)$, $u(t_n)$ 的近似解表示为 u^n. 方程 (1) 的 RAW 滤波蛙跳 (LF) 格式为

$$\begin{aligned}
u^{n+1}&=\tilde{\tilde{u}}^{n-1}+2i\omega\Delta t\tilde{u}^n, &&\text{(LF)}\\
\tilde{\tilde{u}}^{n}&=\tilde{u}^n+\frac{\tau\alpha}{2}(\tilde{\tilde{u}}^{n-1}-2\tilde{u}^n+u^{n+1}), &&\text{(RA)}\\
\tilde{u}^{n+1}&=u^{n+1}+\frac{\tau(\alpha-1)}{2}(\tilde{\tilde{u}}^{n-1}-2\tilde{u}^n+u^{n+1}). &&\text{(W)}
\end{aligned}$$

其中: 无量纲参数 τ 通常为 $\mathcal{O}(0.01\sim0.2)$, 参数 α 通常在 0.5 左右. 这里的 u、$\tilde{u}$ 和 $\tilde{\tilde{u}}$ 分别表示未滤波、一次滤波和两次滤波的值. 当 $\alpha=1$ 时, W 步骤退出, LF-RAW 恢复到 LF-RA. 当 $\tau=0$ 时, 恢复到 LF. 方程 (1) 的 hoRA 时间滤波器的蛙跳格式为

$$\begin{aligned}
\tilde{u}^{n+1}&=u^{n-1}+2i\omega\Delta t\tilde{u}^n, &&\text{(LF)}\\
u^{n}&=\tilde{u}^n+\frac{\tau}{2}(\tilde{u}^{n+1}-2\tilde{u}^n+u^{n-1})-\frac{\tau}{2}(\tilde{u}^n-2u^{n-1}+u^{n-2}). &&\text{(hoRA)}
\end{aligned}$$

其中: 无量纲参数 τ 在区间 (0,1) 内. 这里的 $\tilde{u}$ 和 u 分别表示未滤波和一次滤波的值. 在良好的时间分辨率限制下, 即 $\omega\Delta t\ll1$, LF-hoRA 格式为振幅和相位速度提供了 4 阶的精度.

随后, Layton 等将时间滤波器应用在 Crank-Nicolson-Leapfrog 格式中, 并进行了稳定性分析[27−28]. 给定 $N\times N$ 的矩阵 $\boldsymbol{A}$ 和 $\boldsymbol{\Lambda}$, 考虑下面的系统, 对于任意的 $u:[0,\infty)\to\mathbb{R}^N$

$$\begin{cases}\dfrac{\mathrm{d}u}{\mathrm{d}t}+\boldsymbol{A}u=\boldsymbol{\Lambda}u, \quad u(0)=u_0,\\ \boldsymbol{A}+\boldsymbol{A}^T\geq0, \quad \boldsymbol{\Lambda}=-\boldsymbol{\Lambda}^T.\end{cases}$$

当 $\boldsymbol{A}+\boldsymbol{A}^T=0$ 时, $\boldsymbol{A}=-\boldsymbol{A}^T$, 这个系统是完全守恒的: $|u(t)|=|u(0)|$. 当 $\boldsymbol{A}+\boldsymbol{A}^T>0$ 时, 该系统是耗散的, 即当 $t\to\infty$ 时, $|u(t)|\to 0$. 采用 RAW 滤波器[4,5,29]的 CNLF 方法如下. 给定时间步长 Δt 和起始值 u^0, v^1, 存在 u^n, w^{n+1}, v^{n+1} 满足

$$\frac{u^{n+1}-\tilde{\tilde{u}}^{n-1}}{2\Delta t}=-\boldsymbol{A}\frac{u^{n+1}+\tilde{\tilde{u}}^{n-1}}{2}+\boldsymbol{\Lambda}\tilde{u}^n, \tag{CNLF}$$

$$\tilde{\tilde{u}}^n=\tilde{u}^n+\frac{\tau\alpha}{2}(u^{n+1}-2\tilde{u}^n+\tilde{\tilde{u}}^{n-1}), \tag{RA}$$

$$\tilde{u}^{n+1}=u^{n+1}+\frac{\tau(\alpha-1)}{2}(u^{n+1}-2\tilde{u}^n+\tilde{\tilde{u}}^{n-1}). \tag{W}$$

其中: $\tau\in[0,1]$ 是 RA 滤波器的参数, 通常为 $\mathcal{O}(0.01\sim 0.2)$, α 是 Williams 滤波器参数, 大约为 0.5. 如果 $\alpha=1$, 则 W 步骤退出, 该方法简化为使用 RA 滤波器的 CNLF 方法. 如果 $\tau=0$, 则恢复未滤波的 CNLF 格式. 未经滤波的 CNLF 方法在时间步长条件下 $\Delta t\|\boldsymbol{\Lambda}\|<1$ 是稳定的. 当 $\alpha=1$ 时, 得到的 RA 滤波器的 CNLF 方法在时间步长条件下 $\Delta t\|\boldsymbol{\Lambda}\|\le 1-(\tau/2)$ 是稳定的, $\tau\in(0,1]$. 具有 RAW 滤波器的蛙跳格式是稳定的, 当且仅当时间步长条件满足

$$\Delta t\|\boldsymbol{\Lambda}\|\le\sqrt{2\alpha-1}\sqrt{\frac{2-\tau}{\alpha^2(2+\tau-2\alpha\tau)}}\left(1-\frac{\tau}{2}(2\alpha-1)\right),$$

其中: $\tau\in(0,1]$, $\alpha\in[(1/2),1]$.

2 一些基于时间滤波器的数值方法

这一节中, 我们介绍了时间滤波器在通常数值方法的基础上, 得到的一些数值方法. 包括基于隐式欧拉方法、能量-动量-角动量守恒 (EMAC) 公式、半隐式 DLN 方法、变时间步长变阶的加罚方法等. 考虑速度场和压力空间

$$X:=\{\boldsymbol{v}\in L^2(\Omega)^d:\nabla\boldsymbol{v}\in L^2(\Omega)^{d\times d},\ \boldsymbol{v}=0 \text{ 在 } \partial\Omega \text{ 上}\},$$

$$Q:=\{q\in L^2(\Omega):\int_\Omega q\mathrm{d}\boldsymbol{x}=0\}.$$

其中: $L^p(\Omega)$, $1\le p\le\infty$ 表示勒贝格空间, d 为维数, $H^k(\Omega)$ 为 Sobolev 空间. 对于空间离散化, 用 $X_h\subset X$ 和 $Q_h\subset Q$ 表示速度和压力的有限元空间. 假设空间 (X_h,Q_h) 满足有限元近似稳定性所要求的离散 inf-sup 条件.

2.1 向后欧拉的时间滤波器方法

2018 年, Guzel 等在向后欧拉方法中添加了一个简单的时间滤波器[14]. 考虑初值问题

$$u'(t)=f(t,u(t)),\quad t>0,\quad u(0)=u_0 \tag{2}$$

用 Δt_n 表示第 n 个时间步长. 设 $t_{n+1}=t_n+\Delta t_n$, $\omega_n=\Delta t_n/\Delta t_{n-1}$, τ 为算法参数, u^n 为 $u(t_n)$ 的近似值. 通过标准的向后欧拉 (完全隐式) 方法离散, 然后用一个简单的时间滤波器 (常数时间步) 得到

步骤 1: $\frac{\tilde{u}^{n+1}-u^n}{\Delta t}=f(t_{n+1},\tilde{u}^{n+1})$,

步骤 2: $u^{n+1}=\tilde{u}^{n+1}-\frac{\tau}{2}\{\tilde{u}^{n+1}-2u^n+u^{n-1}\}$.

其中: $\tilde{u}^{n+1}$ 和 u^{n+1} 分别表示未滤波值和滤波值. 上述格式对于 $-2\le\tau<2$ 是 0 稳定的; 对于 $2/3\le\tau\le 2/3$ 是 $\boldsymbol{A}$ 稳定的.

由于向后欧拉是一步方法, 将它扩展成变时间步长格式是明确的. 因此, 实现该方法的关键是将第二步的时间滤波器扩展到变时间步长. 扩展之前首先要定义离散曲率. 设 t_{n-1}, t_n, t_{n+1} 的标准拉格朗日基函数为 $\ell_{n-1}(t)$, $\ell_n(t)$, $\ell_{n+1}(t)$, 经过点 (t_{n-1},u^{n-1}), (t_n,u^n) 和 (t_{n+1},u^{n+1}) 的二次插值多项式 ϕ 是

$$\phi(t)=u^{n+1}\ell_{n+1}(t)+u^n\ell_n(t)+u^{n-1}\ell_{n-1}(t).$$

定义在 (t_{n-1},u^{n-1}), (t_n,u^n), (t_{n+1},u^{n+1}) 的离散曲率为

$$\kappa=\Delta t_{n-1}\Delta t_n\phi''=\frac{2}{1+\omega_n}u^{n+1}-2u^n+\frac{2\omega_n}{1+\omega_n}u^{n-1}.$$

从而第二步扩展到变时间步长格式为

$$u^{n+1}=\tilde{u}^{n+1}-\frac{\tau}{2}\left\{\frac{2}{1+\omega_n}\tilde{u}^{n+1}-2u^n+\frac{2\omega_n}{1+\omega_n}u^{n-1}\right\}.$$

因此, 可以得到原初值问题的变时间步长的时间滤波器格式

步骤 $1'$: $\frac{\tilde{u}^{n+1}-u^n}{\Delta t_n}=f(t_{n+1},\tilde{u}^{n+1})$,

步骤 $2'$: $u^{n+1}=\tilde{u}^{n+1}-\frac{\tau}{2}\left\{\frac{2}{1+\omega_n}\tilde{u}^{n+1}-2u^n+\frac{2\omega_n}{1+\omega_n}u^{n-1}\right\}$.

2.2 向后欧拉 Adams-Bashforth 算法的时间滤波器方法

文献 [22] 中, Decaria 等使用时间滤波器, 将变时间步长的二阶格式嵌入到与 Adams-Bashforth 结合的向后欧拉变时间步长格式算法中.

$$\begin{cases}\boldsymbol{u}_t+\boldsymbol{u}\cdot\nabla\boldsymbol{u}-\nu\Delta\boldsymbol{u}+\nabla p=\boldsymbol{f}(\boldsymbol{x},t), & \Omega\times(0,T]\\ \nabla\cdot\boldsymbol{u}=0, & \Omega\times(0,T]\\ \boldsymbol{u}=0, & \partial\Omega\times(0,T]\\ \boldsymbol{u}(\boldsymbol{x},0)=\boldsymbol{u}_0(\boldsymbol{x}), & \Omega\end{cases}\tag{3}$$

其中: $\Omega\subset\mathbb{R}^d$, $d=2,3$, 表示边界为 $\partial\Omega$ 的一个规则开域, $(0,T]$ 表示时间区间. $\boldsymbol{u}(\boldsymbol{x},t)$ 和 $p(\boldsymbol{x},t)$ 分别是流体的速度和压力, ν, $\boldsymbol{f}$, $\boldsymbol{u}_0$ 分别为流体黏性系数、外力和初速度.

给定 $\boldsymbol{u}_h^n$ 和 $\boldsymbol{u}_h^{n-1}$, 存在 $(\tilde{\boldsymbol{u}}_h^{n+1},p_h^{n+1})$ 满足

$$\left(\frac{\tilde{\boldsymbol{u}}_h^{n+1}-\boldsymbol{u}_h^n}{\Delta t_n},\boldsymbol{v}_h\right)+\nu(\nabla\tilde{\boldsymbol{u}}_h^{n+1},\nabla\boldsymbol{v}_h)+b(E^{n+1}(\boldsymbol{u}_h),E^{n+1}(\boldsymbol{u}_h),\boldsymbol{v}_h)-(p_h^{n+1},\nabla\cdot\boldsymbol{v}_h)=(\boldsymbol{f}^{n+1},\boldsymbol{v}_h),\qquad\forall\ \boldsymbol{v}_h\in X_h,$$

$$(\nabla\cdot\tilde{\boldsymbol{u}}_h^{n+1},q_h)=0,\qquad\forall\ q_h\in Q_h,$$

然后, 通过下式计算 $\boldsymbol{u}_h^{n+1}$,

$$\boldsymbol{u}_h^{n+1}=\tilde{\boldsymbol{u}}_h^{n+1}-\frac{\omega_n}{2\omega_n+1}(\tilde{\boldsymbol{u}}_h^{n+1}-E^{n+1}(\boldsymbol{u}_h)).$$

其中: $\omega_n=\Delta t_n/\Delta t_{n-1}$, $b(\boldsymbol{u},\boldsymbol{v},\boldsymbol{w}):=0.5(\boldsymbol{u}\cdot\nabla\boldsymbol{v},\boldsymbol{w})-0.5(\boldsymbol{u}\cdot\nabla\boldsymbol{w},\boldsymbol{v}),\forall\ \boldsymbol{u},\boldsymbol{v},\boldsymbol{w}\in\boldsymbol{X}$ 是反对称三线性形式. $\boldsymbol{u}_h^{n+1}$ 的二阶外推为 $E^{n+1}(\boldsymbol{u}_h):=(1+\omega_n)\boldsymbol{u}_h^n-\omega_n\boldsymbol{u}_h^{n-1}$.

2.3 EMAC 公式下 NS 方程模块化的时间滤波器方法

对于 NS 方程 (3), 下面的等式为 EMAC 公式的核心思想[30]

$$(\boldsymbol{u}\cdot\nabla)\boldsymbol{u}=2\mathbb{D}(\boldsymbol{u})\boldsymbol{u}-\frac{1}{2}\nabla|\boldsymbol{u}|^2,$$

对于足够光滑的速度场 $\boldsymbol{u}$, 用 $\mathbb{D}(\boldsymbol{u})=(\nabla\boldsymbol{u}+(\nabla\boldsymbol{u})^T)/2$ 表示 $\nabla\boldsymbol{u}$ 的对称部分, 即速度变形张量. EMAC 公式的三线性形式定义为

$$c(\boldsymbol{u},\boldsymbol{v},\boldsymbol{w})=2(\mathbb{D}(\boldsymbol{u})\boldsymbol{v},\boldsymbol{w})+((\nabla\cdot\boldsymbol{u})\boldsymbol{v},\boldsymbol{w}),\quad\forall\ \boldsymbol{u},\boldsymbol{v},\boldsymbol{w}\in\boldsymbol{X}.$$

给定 $\boldsymbol{u}_0$ 的节点插值为 $\boldsymbol{u}_h^0$ 和 $\boldsymbol{u}_h^{-1}$. 然后对于任意 $N-1\geq n\geq 0$, 通过以下步骤求解 $(\boldsymbol{u}_h^{n+1},P_h^{n+1})\in(X_h,Q_h)$.

步骤 1: 通过下式计算 $(\tilde{\boldsymbol{u}}_h^{n+1},P_h^{n+1})\in(X_h,Q_h)$,

$$\left(\frac{\tilde{\boldsymbol{u}}_h^{n+1}-\boldsymbol{u}_h^n}{\Delta t},\boldsymbol{v}_h\right)+\nu(\nabla\tilde{\boldsymbol{u}}_h^{n+1},\nabla\boldsymbol{v}_h)+c(\tilde{\boldsymbol{u}}_h^{n+1},\tilde{\boldsymbol{u}}_h^{n+1},\boldsymbol{v}_h)-(P_h^{n+1},\nabla\cdot\boldsymbol{v}_h)=(\boldsymbol{f}^{n+1},\boldsymbol{v}_h),\qquad\forall\ \boldsymbol{v}_h\in X_h,$$

$$(\nabla\cdot\tilde{\boldsymbol{u}}_h^{n+1},q_h)=0,\qquad\forall\ q_h\in Q_h.$$

步骤 2: 通过下式计算 $\boldsymbol{u}_h^{n+1}$,

$$\boldsymbol{u}_h^{n+1}=\tilde{\boldsymbol{u}}_h^{n+1}-\frac{1}{3}(\tilde{\boldsymbol{u}}_h^{n+1}-2\boldsymbol{u}_h^n+\boldsymbol{u}_h^{n-1}).$$

2.4 变时间步长变阶加罚方法的时间滤波器方法

在不可压缩性条件中添加惩罚项对连续性方程进行扰动[24], 并对 NS 方程 (3) 中的非线性项进行显式地反对称化, 得到加罚 NS 方程

$$\boldsymbol{u}_t-\nu\Delta\boldsymbol{u}+\boldsymbol{u}\cdot\nabla\boldsymbol{u}+\frac{1}{2}(\nabla\cdot\boldsymbol{u})\boldsymbol{u}+\nabla p=\boldsymbol{f},$$

$$\nabla\cdot\boldsymbol{u}+\epsilon p=0.$$

第二式化简得 $p=-(1/\epsilon)\nabla\cdot\boldsymbol{u}$, 其中 $0<\epsilon\ll1$. 将其带入第一个式子中, 得到关于 $\boldsymbol{u}$ 的方程

$$\boldsymbol{u}_{\epsilon,t}-\nu\Delta\boldsymbol{u}_\epsilon+\boldsymbol{u}_\epsilon\cdot\nabla\boldsymbol{u}_\epsilon+\frac{1}{2}(\nabla\cdot\boldsymbol{u}_\epsilon)\boldsymbol{u}_\epsilon-\nabla(\frac{1}{\epsilon}\nabla\cdot\boldsymbol{u}_\epsilon)=\boldsymbol{f}.$$

考虑上面方程的变时间步长的向后欧拉离散格式. 设 Δt_n 为第 $n+1$ 个时间步长, ϵ_n 为第 $n+1$ 个时间步长的参数 ϵ, $t_0=0$, $t_{n+1}=t_n+\Delta t_n$, 则有

$$\frac{\boldsymbol{u}^{n+1}-\boldsymbol{u}^n}{\Delta t_n}+\boldsymbol{u}^*\cdot\nabla\boldsymbol{u}^{n+1}+\frac{1}{2}(\nabla\cdot\boldsymbol{u}^*)\boldsymbol{u}^{n+1}-\nu\Delta\boldsymbol{u}^{n+1}-\nabla(\frac{1}{\epsilon_{n+1}}\nabla\cdot\boldsymbol{u}^{n+1})=\boldsymbol{f}^{n+1},$$

其中: $\boldsymbol{u}^*=(1+(\Delta t_n/\Delta t_{n-1}))\boldsymbol{u}^n-(\Delta t_n/\Delta t_{n-1})\boldsymbol{u}^{n-1}$ 表示 $\boldsymbol{u}$ 在 t_{n+1} 时的标准 (二阶) 线性外推.

给定 $\alpha=2$, $\boldsymbol{u}^n$, $\boldsymbol{u}^{n-1}$, ϵ_{n+1}, ϵ_n 和 ϵ 的容许误差 $TOL=10^{-6}$, 其下界为 $\min TOL=TOL/10$. ϵ 的下界为 $\epsilon_{\min}=10^{-8}$, 上界为 $\epsilon_{\max}=10^{-5}$. 时间步长的容许误差 $tTOL=10^{-5}$, 其下界为 $\min tTOL=tTOL/10$. 计算 $\omega_n=\Delta t_n/\Delta t_{n-1}$ 和

$$\alpha_1=\frac{\omega_n(1+\omega_n)}{1+2\omega_n},\quad \alpha_2=\frac{\omega_n(\omega_{n+1}\omega_n+\omega_n+1)(4\omega_{n+1}^3+5\omega_{n+1}^2+\omega_{n+1})}{3(\omega_n\omega_{n+1}^2+4\omega_n\omega_{n+1}+2\omega_{n+1}+\omega_n+1)}.$$

设 $\boldsymbol{u}^*=(1+\omega_n)\boldsymbol{u}^n-\omega_n\boldsymbol{u}^{n-1}$, 用下式求解 $\tilde{\boldsymbol{u}}^{n+1}$,

$$\frac{\tilde{\boldsymbol{u}}^{n+1}-\boldsymbol{u}_n}{\Delta t_n}+\boldsymbol{u}^*\cdot\nabla\tilde{\boldsymbol{u}}^{n+1}+\frac{1}{2}(\nabla\cdot\boldsymbol{u}^*)\tilde{\boldsymbol{u}}^{n+1}-\nabla(\frac{1}{\epsilon_{n+1}}\nabla\cdot\tilde{\boldsymbol{u}}^{n+1})-\nu\Delta\tilde{\boldsymbol{u}}^{n+1}=\boldsymbol{f}^{n+1}.$$

计算时间步长和 ϵ 以及 D_2 的估计量, 并应用时间滤波器得

$$\begin{aligned}
D_2(n+1)&=\frac{2\Delta t_{n-1}}{\Delta t_{n-1}+\Delta t_n}\tilde{\boldsymbol{u}}^{n+1}-2\boldsymbol{u}^n+\frac{2\Delta t_n}{\Delta t_{n-1}+\Delta t_n}\boldsymbol{u}^{n-1},\quad \boldsymbol{u}^{n+1}=\tilde{\boldsymbol{u}}^{n+1}-\frac{\alpha_1}{2}D_2(n+1),\\
EST_e(n+1)&=\frac{\|\nabla\cdot\boldsymbol{u}^{n+1}\|}{\|\nabla\boldsymbol{u}^{n+1}\|},\quad tEST_1(n+1)=\frac{\alpha_1}{2}\|D_2(n+1)\|,\\
tEST_2(n+1)&=\frac{\alpha_2}{6}\left\|\frac{3\Delta t_{n-2}}{\Delta t_n+\Delta t_{n-1}+\Delta t_{n-2}}D_2(n+1)-\frac{3\Delta t_n}{\Delta t_n+\Delta t_{n-1}+\Delta t_{n-2}}D_2(n)\right\|.
\end{aligned}$$

调整 ϵ 和时间步长的策略: 如果 $EST_e(n+1)>TOL$ 或者 $\min\{tEST_1(n+1),\ tEST_2(n+1)\}>tTOL$, 则有

$$\epsilon_{n+1}\leftarrow\max\{(1-\alpha\Delta t_n)\epsilon_{n+1},0.5\epsilon_{n+1},\epsilon_{\min}\};\quad STEP1=\max\left\{0.9\Delta t_{n-1}\left(\frac{tTOL}{tEST_1(n+1)}\right)^{\frac{1}{2}},0.5\Delta t_n\right\};$$

$$STEP2=\max\left\{0.9\Delta t_{n-1}\left(\frac{tTOL}{tEST_2(n+1)}\right)^{\frac{1}{3}},0.5\Delta t_n\right\};\quad \Delta t_n\leftarrow\max\{STEP1,STEP2\};$$

重复上面步骤.

如果 $EST_e(n+1)<\min TOL$ 或者 $\min\{tEST_1(n+1),\ tEST_2(n+1)\}<\min tTOL$, 则有

$$\epsilon_{n+2}\leftarrow\min\{2\epsilon_{n+1},\epsilon_{\max}\};\quad STEP1\leftarrow\max\left\{\min\left\{0.9\Delta t_n\left(\frac{tTOL}{tEST_1(n+1)}\right)^{\frac{1}{2}},2\Delta t_n\right\},0.5\Delta t_n\right\};$$

$$STEP2\leftarrow\max\left\{\min\left\{0.9\Delta t_n\left(\frac{tTOL}{tEST_2(n+1)}\right)^{\frac{1}{3}},2\Delta t_n\right\},0.5\Delta t_n\right\};\quad \Delta t_{n+1}\leftarrow\max\{STEP1,STEP2\};$$

继续以上步骤.

2.5 重构 DLN 的变时间步长自适应算法时间滤波器方法

Layton 等提出了 DLN 的变步长重构是无条件稳定方法[31]. 下面考虑一个初值问题 (2) 的全隐式欧拉方法

$$\frac{u^{\text{new}}-u^{\text{old}}}{t_{\text{new}}-t_{\text{old}}}=f(t_{\text{new}},u^{\text{new}}). \tag{BE}$$

在 $\{t_n\}_{n\geq 0}$ 时刻, 时间步长 $\Delta t_n=t_{n+1}-t_n$, 令 $\varepsilon_n=(\Delta t_n-\Delta t_{n-1})/(\Delta t_n+\Delta t_{n-1})\in(-1,1)$, $\delta\in[0,1]$ 为任意参数. "单支 (one-leg)" 一词是 Dahlquist 在 1975 年创造的, 用来命名每一步只涉及一个 f 值的多步方法. 作为一种单支的变时间步长 DLN 方法为

$$\frac{\alpha_2u^{n+1}+\alpha_1u^n+\alpha_0u^{n-1}}{\widehat{\Delta t}_n}=f\left(\beta_2^{(n)}t_{n+1}+\beta_1^{(n)}t_n+\beta_0^{(n)}t_{n-1},\beta_2^{(n)}u^{n+1}+\beta_1^{(n)}u^n+\beta_0^{(n)}u^{n-1}\right), \tag{DLN}$$

其系数为

$$\begin{bmatrix}\alpha_2\\ \alpha_1\\ \alpha_0\end{bmatrix}=\begin{bmatrix}\frac{1}{2}(\delta+1)\\ -\delta\\ \frac{1}{2}(\delta-1)\end{bmatrix},\qquad \begin{bmatrix}\beta_2^{(n)}\\ \beta_1^{(n)}\\ \beta_0^{(n)}\end{bmatrix}=\begin{bmatrix}\frac{1}{4}\left(1+\frac{1-\delta^2}{(1+\varepsilon_n\delta)^2}+\varepsilon_n^2\frac{\delta(1-\delta^2)}{(1+\varepsilon_n\delta)^2}+\delta\right)\\ \frac{1}{2}\left(1-\frac{1-\delta^2}{(1+\varepsilon_n\delta)^2}\right)\\ \frac{1}{4}\left(1+\frac{1-\delta^2}{(1+\varepsilon_n\delta)^2}-\varepsilon_n^2\frac{\delta(1-\delta^2)}{(1+\varepsilon_n\delta)^2}-\delta\right)\end{bmatrix}.$$

$\{\alpha_i\}_{i=0}^2$ 与时间步长无关, 而 $\{\beta_i^{(n)}\}_{i=0}^2$ 与时间步长有关. 定义平均时间步长

$$\widehat{\Delta t}_n=\alpha_2\Delta t_n-\alpha_0\Delta t_{n-1}=\delta\frac{\Delta t_n-\Delta t_{n-1}}{2}+\frac{\Delta t_n+\Delta t_{n-1}}{2}.$$

当 $\delta=1$ 时, DLN 方法就变成了隐式中点方法

$$\frac{u^{n+1}-u^n}{\Delta t_n}=f\left(\frac{1}{2}(t_{n+1}+t_n),\frac{1}{2}(u^{n+1}+u^n)\right), \tag{单步中点法}$$

而当 $\delta=0$ 时, DLN 方法是双时间步长的隐式中点方法

$$\frac{u^{n+1}-u^{n-1}}{\Delta t_n+\Delta t_{n-1}}=f\left(\frac{1}{2}(t_{n+1}+t_{n-1}),\frac{1}{2}(u^{n+1}+u^{n-1})\right). \tag{两步中点法}$$

重构 DLN 方法的关键是定义重构系数

$$a_1^{(n)}=\beta_1^{(n)}-\frac{\alpha_1\beta_2^{(n)}}{\alpha_2},\quad a_0^{(n)}=1-a_1^{(n)},\quad b^{(n)}=\frac{\beta_2^{(n)}}{\alpha_2},\quad c_2^{(n)}=\frac{1}{\beta_2^{(n)}},\quad c_1^{(n)}=-\frac{\beta_1^{(n)}}{\beta_2^{(n)}},\quad c_0^{(n)}=-\frac{\beta_0^{(n)}}{\beta_2^{(n)}}.$$

由于 $\alpha_0+\alpha_1+\alpha_2=0$, $\beta_0^{(n)}+\beta_1^{(n)}+\beta_2^{(n)}=1$, 则系数 $a_i^{(n)}$, $b^{(n)}$, $c_i^{(n)}$ 满足 $a_0^{(n)}+a_1^{(n)}=1$, $c_2^{(n)}+c_1^{(n)}+c_0^{(n)}=1$. 该 DLN 方法是无条件 G 稳定的.

下面给出 NS 方程 (3) 的半隐式自适应 DLN 算法[23]. 给定两个之前的解 $\boldsymbol{u}_h^n$, $\boldsymbol{u}_h^{n-1}\in X_h$, p_h^n, $p_h^{n-1}\in Q_h$, 存在 $\boldsymbol{u}_h^{n+1}\in X_h$ 和 $p_h^{n+1}\in Q_h$, 使得所有 $(\boldsymbol{v}_h,q_h)\in X_h\times Q_h$ 满足

$$\begin{aligned}&\left(\frac{\alpha_2\boldsymbol{u}_h^{n+1}+\alpha_1\boldsymbol{u}_h^n+\alpha_0\boldsymbol{u}_h^{n-1}}{\widehat{\Delta t}_n},\boldsymbol{v}_h\right)+\nu(\nabla\boldsymbol{u}_h^{n,\beta},\nabla\boldsymbol{v}_h)+b(\boldsymbol{u}_h^{n,*},\boldsymbol{u}_h^{n,\beta},\boldsymbol{v}_h)-(p_h^{n,\beta},\nabla\cdot\boldsymbol{v}_h)=(\boldsymbol{f}^{n,\beta},\boldsymbol{v}_h)\\&(\nabla\cdot\boldsymbol{u}_h^{n,\beta},q_h)=0\end{aligned} \tag{4}$$

其中: 定义序列 $\{\gamma^n\}_{n=0}^{\infty}$, $\gamma=\boldsymbol{u}_h$, p_h, $\boldsymbol{f}$, 为了方便起见用下式表示

$$\gamma^{n,\beta}:=\beta_2^{(n)}\gamma^{n+1}+\beta_1^{(n)}\gamma^n+\beta_0^{(n)}\gamma^{n-1},\quad t_{n,\beta}=\beta_2^{(n)}t_{n+1}+\beta_1^{(n)}t_n+\beta_0^{(n)}t_{n-1}.$$

$\boldsymbol{u}_h^{n,\beta}$ 的二阶线性外推为

$$\boldsymbol{u}_h^{n,*}=\beta_2^{(n)}\left[(1+\frac{\Delta t_n}{\Delta t_{n-1}})\boldsymbol{u}_h^n-\frac{\Delta t_n}{\Delta t_{n-1}}\boldsymbol{u}_h^{n-1}\right]+\beta_1^{(n)}\boldsymbol{u}_h^n+\beta_0^{(n)}\boldsymbol{u}_h^{n-1}\approx\boldsymbol{u}_h^{n,\beta}.$$

式 (4) 中的半隐式 DLN 算法可以通过后续的重构过程进行简化.

步骤 1: 预处理

$$\boldsymbol{u}_h^{n,\text{old}}=a_1^{(n)}\boldsymbol{u}_h^n+a_0^{(n)}\boldsymbol{u}_h^{n-1},\quad \widehat{\Delta t}_n^{BE}=b^{(n)}\widehat{\Delta t}_n,$$

$$\boldsymbol{u}_h^{n,*}=\beta_2^{(n)}\left[(1+\frac{\Delta t_n}{\Delta t_{n-1}})\boldsymbol{u}_h^n-\frac{\Delta t_n}{\Delta t_{n-1}}\boldsymbol{u}_h^{n-1}\right]+\beta_1^{(n)}\boldsymbol{u}_h^n+\beta_0^{(n)}\boldsymbol{u}_h^{n-1}.$$

步骤 2: 用半隐式向后欧拉格式求解 $\tilde{\boldsymbol{u}}_h^{n+1}$ 和 $\tilde{p}_h^{n+1}$

$$\left(\frac{\tilde{\boldsymbol{u}}_h^{n+1}-\boldsymbol{u}_h^{n,\text{old}}}{\widehat{\Delta t}_n^{BE}},\boldsymbol{v}_h\right)+b(\boldsymbol{u}_h^{n,*},\tilde{\boldsymbol{u}}_h^{n+1},\boldsymbol{v}_h)+\nu(\nabla\tilde{\boldsymbol{u}}_h^{n+1},\nabla\boldsymbol{v}_h)-(\tilde{p}_h^{n+1},\nabla\cdot\boldsymbol{v}_h)=(\boldsymbol{f}^{n,\beta},\boldsymbol{v}_h),$$

$$(\nabla\cdot\tilde{\boldsymbol{u}}_h^{n+1},q_h)=0.$$

步骤 3: 后处理

$$\boldsymbol{u}_h^{n+1}=c_2^{(n)}\tilde{\boldsymbol{u}}_h^{n+1}+c_1^{(n)}\boldsymbol{u}_h^n+c_0^{(n)}\boldsymbol{u}_h^{n-1},\quad p_h^{n+1}=c_2^{(n)}\tilde{p}_h^{n+1}+c_1^{(n)}p_h^n+c_0^{(n)}p_h^{n-1}.$$

接下来, 定义局部截断误差并给出重构 DLN 的自适应算法. 给定四个解 $\boldsymbol{u}_h^n$, $\boldsymbol{u}_h^{n-1}$, $\boldsymbol{u}_h^{n-2}$, $\boldsymbol{u}_h^{n-3}$, NS 方程在 t_{n+1} 时刻的数值解为

$$\begin{aligned}\boldsymbol{u}_{h,AB2}^{n+1}=&\left[1+\alpha_2\frac{(t_{n+1}-t_n)(t_{n+1}+t_n-2t_{n-1,\beta})}{2(t_{n,\beta}-t_{n-1,\beta})\widehat{\Delta t}_{n-1}}\right]\boldsymbol{u}_h^n\\&+\frac{(t_{n+1}-t_n)}{2(t_{n,\beta}-t_{n-1,\beta})}\left[\alpha_1\frac{(t_{n+1}+t_n-2t_{n-1,\beta})}{\widehat{\Delta t}_{n-1}}-\alpha_2\frac{(t_{n+1}+t_n-2t_{n,\beta})}{\widehat{\Delta t}_{n-2}}\right]\boldsymbol{u}_h^{n-1}\\&+\frac{(t_{n+1}-t_n)}{2(t_{n,\beta}-t_{n-1,\beta})}\left[\alpha_0\frac{(t_{n+1}+t_n-2t_{n-1,\beta})}{\widehat{\Delta t}_{n-1}}-\alpha_1\frac{(t_{n+1}+t_n-2t_{n,\beta})}{\widehat{\Delta t}_{n-2}}\right]\boldsymbol{u}_h^{n-2}\\&-\alpha_0\frac{(t_{n+1}-t_n)(t_{n+1}+t_n-2t_{n,\beta})}{2(t_{n,\beta}-t_{n-1,\beta})\widehat{\Delta t}_{n-2}}\boldsymbol{u}_h^{n-3}\end{aligned}\tag{5}$$

记估计子为

$$\widehat{T}_{n+1}=\frac{|G^{(n)}|}{|G^{(n)}+R^{(n)}|}\|\boldsymbol{u}_{h,DLN}^{n+1}-\boldsymbol{u}_{h,AB2}^{n+1}\|,\qquad\text{(绝对估计子)}$$

$$\widehat{T}_{n+1}=\frac{|G^{(n)}|}{|G^{(n)}+R^{(n)}|}\frac{\|\boldsymbol{u}_{h,DLN}^{n+1}-\boldsymbol{u}_{h,AB2}^{n+1}\|}{\|\boldsymbol{u}_{h,DLN}^{n+1}\|}.\qquad\text{(相对估计子)}$$

其中:

$$\begin{aligned}G^{(n)}=&\left(\frac{1}{2}-\frac{\alpha_0}{2\alpha_2}\frac{1-\varepsilon_n}{1+\varepsilon_n}\right)\left(\beta_2^{(n)}-\beta_0^{(n)}\frac{1-\varepsilon_n}{1+\varepsilon_n}\right)^2+\frac{\alpha_0}{6\alpha_2}\left(\frac{1-\varepsilon_n}{1+\varepsilon_n}\right)^3-\frac{1}{6}\\R^{(n)}=&\frac{1}{12}\left[2+\frac{3(1-\varepsilon_n)}{1+\varepsilon_n}\left(1-\beta_2^{(n-2)}\frac{1-\varepsilon_{n-1}}{1+\varepsilon_{n-1}}+\beta_0^{(n-2)}\frac{1-\varepsilon_{n-2}}{1+\varepsilon_{n-2}}\frac{1-\varepsilon_{n-1}}{1+\varepsilon_{n-1}}\right)\right.\\&\times\left(1-\beta_2^{(n-1)}\frac{1-\varepsilon_n}{1+\varepsilon_n}+\beta_0^{(n-1)}\frac{1-\varepsilon_{n-1}}{1+\varepsilon_{n-1}}\frac{1-\varepsilon_n}{1+\varepsilon_n}\right)\\&+\frac{3(1-\varepsilon_n)}{1+\varepsilon_n}\left(\frac{2}{1+\varepsilon_n}-\beta_2^{(n-2)}\frac{1-\varepsilon_{n-1}}{1+\varepsilon_{n-1}}\frac{1-\varepsilon_n}{1+\varepsilon_n}+\beta_0^{(n-2)}\frac{1-\varepsilon_{n-2}}{1+\varepsilon_{n-2}}\frac{1-\varepsilon_{n-1}}{1+\varepsilon_{n-1}}\frac{1-\varepsilon_n}{1+\varepsilon_n}\right)\\&\left.\times\left(-\beta_2^{(n-1)}+\beta_0^{(n-1)}\frac{1-\varepsilon_{n-1}}{1+\varepsilon_{n-1}}\right)\right]\end{aligned}\tag{6}$$

重构 DLN 的自适应算法. 给定容许误差 TOL, $\boldsymbol{u}_h^n$, $\boldsymbol{u}_h^{n-1}$, $\boldsymbol{u}_h^{n-2}$, $\boldsymbol{u}_h^{n-3}$ 和 p_h^n, p_h^{n-1}, p_h^{n-2}, p_h^{n-3}. 当前时间步长为 Δt_n, 前三个时间步长为 Δt_{n-1}, Δt_{n-2}, Δt_{n-3}. 为了最大限度地减少步长拒绝的次数, 选择安全系数 $\kappa\in(0,1]$.

通过式 (4) 计算 DLN 解 $\boldsymbol{u}_{h,DLN}^{n+1}$ 和 $p_{h,DLN}^{n+1}$; 通过式 (5) 计算数值解 $\boldsymbol{u}_{h,AB2}^{n+1}$. 接着用 Δt_n, Δt_{n-1}, Δt_{n-2}, Δt_{n-3} 更新 ε_n, ε_{n-1}, ε_{n-2}. 通过式 (6) 计算 $G^{(n)}$, $R^{(n)}$; 再计算估计子

$$\widehat{T}_{n+1}\Leftarrow\frac{|G^{(n)}|}{|G^{(n)}+R^{(n)}|}\|\boldsymbol{u}_{h,DLN}^{n+1}-\boldsymbol{u}_{h,AB2}^{n+1}\|,$$

或

$$\widehat{T}_{n+1} \Leftarrow \frac{|G^{(n)}|}{|G^{(n)}+R^{(n)}|}\frac{\|\boldsymbol{u}_{h,DLN}^{n+1}-\boldsymbol{u}_{h,AB2}^{n+1}\|}{\|\boldsymbol{u}_{h,DLN}^{n+1}\|}.$$

如果 $\widehat{T}_{n+1}<TOL$, 则 $\boldsymbol{u}_h^{n+1} \Leftarrow \boldsymbol{u}_{h,DLN}^{n+1}$ 和 $p_h^{n+1} \Leftarrow p_{h,DLN}^{n+1}$, 更新时间步长

$$\Delta t_{n+1} \Leftarrow \Delta t_n \cdot \min\left\{1.5, \max\left\{0.2, \kappa\left(\frac{TOL}{\widehat{T}_{n+1}}\right)^{\frac{1}{3}}\right\}\right\},$$

如果 $\widehat{T}_{n+1} \geq TOL$, 则重新计算当前时间步长

$$\Delta t_{n} \Leftarrow \Delta t_n \cdot \min\left\{1.5, \max\left\{0.2, \kappa\left(\frac{TOL}{\widehat{T}_{n+1}}\right)^{\frac{1}{3}}\right\}\right\}.$$

3　求解几种偏微分方程的时间滤波器方法

本节中, 我们介绍了利用时间滤波器求解三种偏微分方程. 当然, 时间滤波器被广泛地运用于许多偏微分方程, 引言中也做了一部分总结, 这里只介绍其中三类方程.

3.1　具有斜率选择的分子束外延 (MBE) 方程的能量稳定时间滤波向后欧拉方法

考虑在 $\Omega \subset \mathbb{R}^2$ 上的 MBE 方程[32], 其中 $0 \leq t \leq T$,

$$\partial_t \Phi = -\kappa\mu, \quad \mu := \frac{\delta E}{\delta \Phi} = \varepsilon^2 \Delta^2 \Phi - \nabla \cdot f(\nabla \Phi),$$

其中: 迁移率系数 $\kappa>0$, 界面宽度参数 $\varepsilon>0$, Φ 是周期性高度函数, $f(\boldsymbol{v}) := (|\boldsymbol{v}|^2-1)\boldsymbol{v}$ 是非线性体积力. 向后欧拉方法为: 给定时间步长 Δt, 时间离散格式为

$$\frac{\phi^n-\phi^{n-1}}{\Delta t} = -\kappa\mu^n, \quad \mu^n := \varepsilon^2\Delta^2\phi^n - \nabla\cdot f(\nabla\phi^n), \quad n \geq 1.$$

如果再添加一行额外的代码, 可以得到以下 MBE 方程的向后欧拉时间滤波器格式.

步骤 1: 通过向后欧拉格式计算未滤波的解 $\tilde{\phi}^n$,

$$\frac{1}{\Delta t}(\tilde{\phi}^n-\phi^{n-1}) = -\kappa\tilde{\mu}^n, \quad \tilde{\mu}^n := \varepsilon^2\Delta^2\tilde{\phi}^n - \nabla\cdot f(\nabla\tilde{\phi}^n), \quad n \geq 2.$$

步骤 2: 通过时间滤波器计算解 ϕ^n,

$$\phi^n = \tilde{\phi}^n - \frac{1}{3}(\tilde{\phi}^n - 2\phi^{n-1} + \phi^{n-2}), \quad n \geq 2.$$

3.2　抛物型方程的变时间步长滤波向后欧拉方法

考虑以下有界凸域 Ω 中的反应扩散方程[33], 其中 $0 \leq t \leq T$,

$$\partial_t u = \mathscr{L}u + f(t,x), \quad x \in \Omega.$$

在光滑边界 $\partial\Omega$ 上满足狄利克雷边界条件 $u=0$, 初始值由 $u(0,x)=u_0$ 给出. 空间算子 $\mathscr{L}$ 定义为

$$\mathscr{L}u := \varepsilon^2\Delta u - \kappa(x)u,$$

其中: 扩散系数 $\varepsilon>0$, 反应系数 $\kappa(x)$ 是光滑的且以 $\kappa_0>0$ 为界, 即 $|\kappa(x)| \leq \kappa_0$.

下面给出变时间步长滤波向后欧拉 (FiBE) 格式. 给定时间步长 $\Delta t_n = t_n - t_{n-1}$, $1 \leq n \leq N$, 此外, 相邻时间步长比 $\omega_n = \Delta t_n/\Delta t_{n-1}$, $2 \leq n \leq N$, $\omega_1 = 0$. 滤波参数 $\tau_n = \omega_n(1+\omega_n)/(1+2\omega_n)$.

步骤 1: 通过向后欧拉格式计算未滤波的解 $\tilde{u}^n$,

$$\frac{\tilde{u}^n - u^{n-1}}{\Delta t_n} = \mathscr{L}\tilde{u}^n + f^n, \quad 2 \leq n \leq N.$$

步骤 2: 通过时间滤波过程计算解 u^n,

$$u^n = \tilde{u}^n - \frac{\tau_n}{2}\left[\frac{2}{1+\omega_n}(\tilde{u}^n - u^{n-1}) - \frac{2\omega_n}{1+\omega_n}(u^{n-1}-u^{n-2})\right], \quad 2 \leq n \leq N.$$

通常, 向后欧拉格式产生一阶精度, 而通过添加第二步的时间滤波过程, FiBE 格式在时间上是二阶精度.

3.3 受 DNA 转录模型启发的一个线性双曲方程的时间滤波方法

生物模型中, 假设分子马达在链上的密度 $\rho(x,t)$ 遵循以下偏微分方程[20]

$$\begin{aligned}\rho_t+\bar{v}\rho_x&=0,\quad 0<x\le 1,\quad t>0\\ \rho(x,0)&=\rho_0\\ \rho(0,t)&=\rho_I\end{aligned}\tag{7}$$

其中: $\bar{v}>0$ 是恒定的延伸速度, ρ_0 和 ρ_I 分别是初始条件和边界条件. 变量 $\rho(x,t)$ 表示在 t 时刻沿着链的空间点 x 处的分子马达密度. 链上核糖体的初始密度由 ρ_0 参数决定. 流入边界在 $x=0$ 处, 代表基因的起始位点或启动子位点. 流出边界记为 $x=1$, 与基因的终止位点相对应.

下面给出与完全显式迎风方法相结合的时间滤波器方法. 该方法是在网格点 $x_j=jh,\ j=0,1,2,\cdots,M$ 且 $t_n=nk,\ n=0,1,2,\cdots,N$ 处离散 PDE 方程. 定义近似值 $P_j^n\approx\rho(x_j,t_n)$, 则方程 (7) 的离散形式为

$$\frac{P_j^{n+1}-P_j^n}{\Delta t}+\bar{v}\frac{(P_j^n-P_{j-1}^n)}{\Delta x}=0,$$

典型的迎风格式为

$$P_j^{n+1}=P_j^n-\mu(P_j^n-P_{j-1}^n).$$

为了简化方法的表达式, 定义参数 $\mu=\bar{v}\Delta t/\Delta x$, 称为 Courant 数, 并且对于一个固定的参数选择, μ 的值是常数. 已知迎风方法在下列情况下是稳定的

$$0<\mu\le 1.$$

迎风格式的完全显式离散化与点 (x_j,t_{n+1}) 处的时间滤波器相结合, 有

步骤 1: $\tilde{P}_j^{n+1}=(1-\mu)P_j^n+\mu P_{j-1}^n$,

步骤 2: $P_j^{n+1}=\tilde{P}_j^{n+1}-\frac{\tau}{2}(\tilde{P}_j^{n+1}-2P_j^n+P_j^{n-1})$.

其中: $\tilde{P}_j^{n+1}$ 和 P_j^{n+1} 分别表示未滤波和滤波后的密度近似. 第二步中的 τ 是时间滤波器参数, $0<\tau<2$. 可以注意到, $\tau=0$ 时该格式简化为经典迎风格式. 由于滤波器是线性的, 整个过程可以简化为单步方法. 将滤波后变量代入迎风格式, 得到单步格式

$$P_j^{n+1}=\left[\tau+(1-\mu)(1-\frac{\tau}{2})\right]P_j^n-\frac{\tau}{2}P_j^{n-1}+\mu(1-\frac{\tau}{2})P_{j-1}^n.$$

4 NS 方程的一种时间变滤波器方法

这一节中, 针对 NS 方程 (3), 本文提出了基于指数型辅助变量的时间变滤波器方法. 选取辅助变量 $Q(t)=\exp(-t/T)$ 来实现无条件稳定性, 同时基于上述讨论, 我们知道利用时间滤波器可以将向后欧拉的一阶精度提高到二阶. 于是, 首先构造了 NS 方程的等价形式

$$\begin{cases}\dfrac{\partial\boldsymbol{u}}{\partial t}+Q(t)\exp\left(\dfrac{t}{T}\right)(\boldsymbol{u}\cdot\nabla)\boldsymbol{u}-\nu\Delta\boldsymbol{u}+\nabla p=\boldsymbol{f},\quad \nabla\cdot\boldsymbol{u}=0,\\ \dfrac{\mathrm{d}Q(t)}{\mathrm{d}t}=-\dfrac{1}{T}Q(t)+\exp\left(\dfrac{t}{T}\right)b(\boldsymbol{u},\boldsymbol{u},\boldsymbol{u}).\end{cases}$$

相对应变分形式可以写成: 对于所有 $t\in(0,T]$, 存在 $(\boldsymbol{u},p)\in X\times Q$, 使得

$$\begin{cases}(\boldsymbol{u}_t,\boldsymbol{v})+Q(t)\exp\left(\dfrac{t}{T}\right)b(\boldsymbol{u},\boldsymbol{u},\boldsymbol{v})+\nu(\nabla\boldsymbol{u},\nabla\boldsymbol{v})-(p,\nabla\cdot\boldsymbol{v})=(\boldsymbol{f},\boldsymbol{v}),\\ (\nabla\cdot\boldsymbol{u},q)=0,\qquad\forall\ (\boldsymbol{v},q)\in X\times Q,\\ \dfrac{\mathrm{d}Q(t)}{\mathrm{d}t}=-\dfrac{1}{T}Q(t)+\exp\left(\dfrac{t}{T}\right)b(\boldsymbol{u},\boldsymbol{u},\boldsymbol{u}).\end{cases}$$

接下来, 定义时间离散格式: 给定 $\boldsymbol{u}_h^n \in X_h$ 和 $Q^n \in \mathbb{R}$, 存在 $(\boldsymbol{u}_h^{n+1}, p_h^{n+1}, Q^{n+1}) \in X_h \times Q_h \times \mathbb{R}$ 满足

$$\begin{aligned}
&\left(\frac{\boldsymbol{u}_h^{n+1}-\boldsymbol{u}_h^n}{\Delta t}, \boldsymbol{v}_h\right)+\nu(\nabla \boldsymbol{u}_h^{n+1}, \nabla \boldsymbol{v}_h)+Q^{n+1}\exp\left(\frac{t_{n+1}}{T}\right)b(\boldsymbol{u}_h^n, \boldsymbol{u}_h^n, \boldsymbol{v}_h)-(p_h^{n+1}, \nabla\cdot\boldsymbol{v}_h)=(\boldsymbol{f}(t_{n+1}), \boldsymbol{v}_h)\\
&(\nabla\cdot\boldsymbol{u}_h^{n+1}, q_h)=0, \qquad \forall\ (\boldsymbol{v}_h, q_h)\in X_h\times Q_h\\
&\frac{Q^{n+1}-Q^n}{\Delta t}=-\frac{1}{T}Q^{n+1}+\exp\left(\frac{t_{n+1}}{T}\right)b(\boldsymbol{u}_h^n, \boldsymbol{u}_h^n, \boldsymbol{u}_h^{n+1})
\end{aligned} \tag{8}$$

最后, 给出全离散时间变滤波器格式. 设 $\{t_n\}_{n=0}^N$, $N>0$ 是 $[0,T]$ 上的非一致剖分, 变时间步长 $\Delta t_n = t_{n+1}-t_n$ 和其最大值 $\Delta t_{\max}=\max\limits_{1\le n\le N}\{\Delta t_n\}$. 时间步长比 $\omega_n=\Delta t_n/\Delta t_{n-1}$, $\omega_{\min}=\min\limits_{1\le n\le N}\{\omega_n\}$, $\omega_{\max}=\max\limits_{1\le n\le N}\{\omega_n\}$, 并且 $\omega_0=1$. 我们假设 $\{\omega_n\}$ 是一致有界序列. 除此之外, 用 $E(\boldsymbol{u}_h^n):=(1+\omega_n)\boldsymbol{u}_h^n-\omega_n\boldsymbol{u}_h^{n-1}$ 来表示 $\boldsymbol{u}_h^{n+1}$ 的二阶外推. 特别的, 记 $E(\boldsymbol{u}_h^0):=\boldsymbol{u}_h^0$.

步骤 1: 给定 $\tilde{\boldsymbol{u}}_h^n, \tilde{\boldsymbol{u}}_h^{n-1}, \boldsymbol{u}_h^n \in X_h$ 和 $Q^{n-1}, Q^n\in\mathbb{R}$, 存在 $(\tilde{\boldsymbol{u}}_h^{n+1}, \tilde{p}_h^{n+1})\in X_h\times Q_h$ 满足

$$\begin{aligned}
&\left(\frac{\tilde{\boldsymbol{u}}_h^{n+1}-\boldsymbol{u}_h^n}{\Delta t_n}, \boldsymbol{v}_h\right)+\nu(\nabla\tilde{\boldsymbol{u}}_h^{n+1}, \nabla\boldsymbol{v}_h)+Q^{n+1}\exp\left(\frac{t_{n+1}}{T}\right)b(E(\tilde{\boldsymbol{u}}_h^n), E(\tilde{\boldsymbol{u}}_h^n), \boldsymbol{v}_h)\\
&\quad+(\nabla\cdot\tilde{\boldsymbol{u}}_h^{n+1}, q_h)-(\tilde{p}_h^{n+1}, \nabla\cdot\boldsymbol{v}_h)=(\boldsymbol{f}(t_{n+1}), \boldsymbol{v}_h)\\
&\frac{\frac{1+2\omega_n}{1+\omega_n}Q^{n+1}-(1+\omega_n)Q^n+\frac{\omega_n^2}{1+\omega_n}Q^{n-1}}{\Delta t_n}=-\frac{1}{T}Q^{n+1}+\exp\left(\frac{t_{n+1}}{T}\right)b(E(\tilde{\boldsymbol{u}}_h^n), E(\tilde{\boldsymbol{u}}_h^n), \tilde{\boldsymbol{u}}_h^{n+1})
\end{aligned} \tag{9}$$

步骤 2: 从步骤 1 中求得 $\tilde{\boldsymbol{u}}_h^{n+1}, \tilde{p}_h^{n+1}$, 通过下式计算 $(\boldsymbol{u}_h^{n+1}, p_h^{n+1})\in X_h\times Q_h$

$$\begin{aligned}
\boldsymbol{u}_h^{n+1}&=\tilde{\boldsymbol{u}}_h^{n+1}-\frac{\omega_n(1+\omega_n)}{2(1+2\omega_n)}\left(\frac{2}{1+\omega_n}\tilde{\boldsymbol{u}}_h^{n+1}-2\boldsymbol{u}_h^n+\frac{2\omega_n}{1+\omega_n}\boldsymbol{u}_h^{n-1}\right),\\
p_h^{n+1}&=\tilde{p}_h^{n+1}-\frac{\omega_n(1+\omega_n)}{2(1+2\omega_n)}\left(\frac{2}{1+\omega_n}\tilde{p}_h^{n+1}-2p_h^n+\frac{2\omega_n}{1+\omega_n}p_h^{n-1}\right).
\end{aligned}$$

下面我们提出了自适应算法.

步骤 1: 给定 Q^0, $\boldsymbol{u}_h^0$ 和 $\Delta t=\Delta t_0$, 求解式 (8) 得到 Q^1 和 $\boldsymbol{u}_h^1$. 然后, 给定容许误差 TOL, Δt_1 和 Δt_2, 依次求解式 (9) 并使用时间滤波器得到 Q^2, $\boldsymbol{u}_h^2$ 和 Q^3, $\boldsymbol{u}_h^3$.

步骤 2: 对于 $n=2,3,\cdots,N-2$, 求解式 (9) 并使用时间滤波器得到 $\boldsymbol{u}_h^{n+2}$ 和 Q^{n+2}, 重复以上步骤.

步骤 3: 自适应时间步长 Δt_n. 计算估计子 EST 和 EST_Q,

$$\begin{aligned}
EST^{n+1}&=\frac{\omega_{n-1}\omega_n(1+\omega_n)}{1+2\omega_n+\omega_{n-1}(1+4\omega_n+3\omega_n^2)}\|\boldsymbol{D}^{n+1}\|_0,\\
EST_Q^{n+1}&=\frac{\omega_{n-1}\omega_n(1+\omega_n)}{1+2\omega_n+\omega_{n-1}(1+4\omega_n+3\omega_n^2)}|D_Q^{n+1}|.
\end{aligned}$$

其中:

$$\begin{aligned}
\boldsymbol{D}^{n+1}&=\boldsymbol{u}_h^{n+1}-\frac{(1+\omega_n)(1+\omega_{n-1}(1+\omega_n))}{1+\omega_{n-1}}\boldsymbol{u}_h^n+\omega_n(1+\omega_{n-1}(1+\omega_n))\boldsymbol{u}_h^{n-1}-\frac{\omega_{n-1}^2\omega_n(1+\omega_n)}{1+\omega_{n-1}}\boldsymbol{u}_h^{n-2},\\
D_Q^{n+1}&=Q^{n+1}-\frac{(1+\omega_n)(1+\omega_{n-1}(1+\omega_n))}{1+\omega_{n-1}}Q^n+\omega_n(1+\omega_{n-1}(1+\omega_n))Q^{n-1}-\frac{\omega_{n-1}^2\omega_n(1+\omega_n)}{1+\omega_{n-1}}Q^{n-2}.
\end{aligned}$$

如果 $\max\{EST^{n+1},\ EST_Q^{n+1}\}<TOL/3$, 则更新时间步长 $\Delta t_{n+1}=0.9\Delta t_n\min\{2,\ (TOL/EST^{n+1})^{1/3}, (\mathrm{TOL}/EST_Q^{n+1})^{1/3}\}$; 如果 $TOL/3\le\max\{EST^{n+1},\ EST_Q^{n+1}\}\le TOL$, 则更新时间步长 $\Delta t_{n+1}=0.9\Delta t_n\min\{1,\ (TOL/EST^{n+1})^{1/3},\ (TOL/EST_Q^{n+1})^{1/3}\}$; 若以上两个条件都不满足, 则返回步骤 2 后重新计算当前时间步长 $\Delta t_n=0.7\Delta t_n\min\{(TOL/EST^{n+1})^{1/3},\ (TOL/EST_Q^{n+1})^{1/3}\}$.

5　结论和展望

本文主要阐述了时间滤波器的发展历程. 通过对时间滤波器的分析, 我们知道大多数求解 PDE 的方法与时间滤波器技术相结合, 可以提高精度. 之所以专注于时间滤波方法, 是因为它是非侵入性的; 也就是说, 它可

以被视为对现有近似方法进行修改的后处理步骤, 而不对原始求解器进行任何更改. 这种方法产生了一种易于实现现有代码的算法, 并且能够以较小的计算成本提高现有求解器的准确性和稳定性. 总的来说, 时间滤波器技术的应用主要是为提高处理过程中的效率和精度, 减少因各种因素引起的误差和损失.

虽然时间滤波器得到了很大发展, 但仍然存在许多挑战和未解决的问题. 比如, 如何进一步提高时间滤波器的精度和效率, 如何处理大规模和复杂的时间序列数据, 如何将其扩展到更复杂的耦合流体模型中, 等等.

参考文献:

[1] ROBERT A J. The integration of a low order spectral form of the primitive meteorological equations[J]. Journal of the Meteorological Society of Japan, 1966, 44(5): 237-245.

[2] ROBERT A J. The integration of a spectral model of the atmosphere by the implicit method[C]//World Meteorological Organization/International Union of Geodesy and Geophysics Symposium on Numerical Weather Prediction. Tokyo: Japan Meteorological Agency, 1969: 19-24.

[3] ASSELIN R. Frequency filter for time integrations[J]. Monthly Weather Review, 1972, 100(6): 487-490.

[4] WILLIAMS P D. A proposed modification to the Robert-Asselin time filter[J]. Monthly Weather Review, 2009, 137(8): 2538-2546.

[5] WILLIAMS P D. The RAW filter: An improvement to the Robert-Asselin filter in semi-implicit integrations[J]. Monthly Weather Review, 2011, 139(6): 1996-2007.

[6] AMEZCUA J, WILLIAMS P D. The composite-tendency Robert-Asselin-Williams (RAW) filter in semi-implicit integrations[J]. Quarterly Journal of the Royal Meteorological Society, 2015, 141(688): 764-773.

[7] LI Y, TRENCHEA C. A higher-order Robert-Asselin type time filter[J]. Journal of Computational Physics, 2014, 259(C): 23-32.

[8] GUZEL A, TRENCHEA C. The Williams step increases the stability and accuracy of the hoRA time filter[J]. Applied Numerical Mathematics, 2018, 131(C): 158-173.

[9] SCHLESINGER R E, UCCELLINI L W, JOHNSON D R. The effects of the Asselin time filter on numerical solutions to the linearized shallow-water wave equations[J]. Monthly Weather Review, 1983, 111(3): 455-467.

[10] DÉQUÉ M, CARIOLLE D. Some destabilizing properties of the Asselin time filter[J]. Monthly Weather Review, 1986, 114(5): 880-884.

[11] CORDERO E, STANIFORTH A. A problem with the Robert-Asselin time filter for three-time-level semi-implicit semi-Lagrangian discretizations[J]. Monthly Weather Review, 2004, 132(2): 600-610.

[12] DECARIA V, LAYTON W, MCLAUGHLIN M. A conservative, second order, unconditionally stable artificial compression method[J]. Computer Methods in Applied Mechanics and Engineering, 2017, 325: 733-747.

[13] GUZEL A, LAYTON W, MCLAUGHLIN M, et al. Time filters and spurious acoustics in artificial compression methods[J]. Numerical Methods for Partial Differential Equations, 2022, 38(6): 1908-1928.

[14] GUZEL A, LAYTON W. Time filters increase accuracy of the fully implicit method[J]. BIT Numerical Mathematics, 2018, 58: 301-315.

[15] QIN Y, HOU Y R. The time filter for the non-stationary coupled Stokes/Darcy model[J]. Applied Numerical Mathematics, 2019, 146(C): 260-275.

[16] CIBIK A, EROGLU F G, KAYA S. Analysis of second order time filtered backward Euler method for MHD equations[J]. Journal of Scientific Computing, 2020, 82(2): 1-25.

[17] VICTOR D, WILLIAM L, ZHAO H Y. A time-accurate, adaptive discretization for fluid flow problems[J]. International Journal of Numerical Analysis and Modeling, 2020, 17(2): 254-280.

[18] QIN Y, WANG Y S, LI J. A variable time step time filter algorithm for the geothermal system[J]. SIAM Journal on Numerical Analysis, 2022, 60(5): 2781-2806.

[19] ZENG Y H, HUANG P Z, HE Y N. A time filter method for solving the double-diffusive natural convection model[J]. Computers & Fluids, 2022, 235: 105265.

[20] BOATMAN K, DAVIS L, PAHLEVANI F, et al. Numerical analysis of a time filtered scheme for a linear hyperbolic equation inspired by DNA transcription modeling[J]. Journal of Computational and Applied Mathematics, 2023, 429: 115135.

[21] LI W, HUANG P Z. On a two-order temporal scheme for Navier-Stokes/Navier-Stokes equations[J]. Applied Numerical Mathematics, 2023, 194: 1-17.

[22] DECARIA V, SCHNEIER M. An embedded variable step IMEX scheme for the incompressible Navier-Stokes equations[J]. Computer Methods in Applied Mechanics and Engineering, 2021, 376: 113661.

[23] PEI W L. The semi-implicit DLN algorithm for the Navier-Stokes equations[J]. Numerical Algorithms, 2024, 97(4): 1673-1713.

[24] KEAN K, XIE X H, XU S X. A doubly adaptive penalty method for the Navier Stokes equations[J]. International Journal of Numerical Analysis and Modeling, 2023, 20(3): 407-436.

[25] WU J L, LI N, FENG X L. Analysis of a filtered time-stepping finite element method for natural convection problems[J]. SIAM Journal on Numerical Analysis, 2023, 61(2): 837-871.

[26] LI Y, WILLIAMS P D. Analysis of the RAW filter in composite-tendency leapfrog integrations[R]. Pittsburgh: University of Pittsburgh, 2014.

[27] HURL N, LAYTON W, LI Y, et al. Stability analysis of the Crank-Nicolson-Leapfrog method with the Rober-Asselin-Williams time filter[J]. BIT Numerical Mathematics, 2014, 54(4): 1009-1021.

[28] LAYTON W, LI Y, TRENCHEA C. Recent developments in IMEX methods with time filters for systems of evolution equations[J]. Journal of Computational and Applied Mathematics, 2016, 299: 50-67.

[29] AMEZCUA J, KALNAY E, WILLIAMS P D. The effects of the RAW filter on the climatology and forecast skill of the SPEEDY model[J]. Monthly Weather Review, 2011, 139(2): 608-619.

[30] DEMIR M, ÇIBIK A, KAYA S. Time filtered second order backward Euler method for EMAC formulation of Navier-Stokes equations[J]. Journal of Mathematical Analysis and Applications, 2022, 516(2): 126562.

[31] LAYTON W, PEI W L, TRENCHEA C. Refactorization of a variable step, unconditionally stable method of Dahlquist, Liniger and Nevanlinna[J]. Applied Mathematics Letters, 2022, 125: 107789.

[32] WANG J X, LIAO H L, ZHAO Y. An energy stable filtered backward Euler scheme for the MBE equation with slope selection[J]. Numerical Mathematics: Theory, Methods and Applications, 2023, 16(1): 165-181.

[33] LIAO H L, TANG T, ZHOU T. Stability and convergence of the variable-step time filtered backward Euler scheme for parabolic equations[J]. BIT Numerical Mathematics, 2023, 63(3): 39.

责任编辑：张自强

(上接第35页)

[13] CHEUNG K C. Optimal reinsurance revisited-A geometric approach[J]. ASTIN Bulletin, 2010, 40(1): 221-239.

[14] ANDERSON E W, HANSEN L P, SARGENT T J. A quartet of semigroups for model specification, robustness, prices of risk, and model detection[J]. Journal of the European Economic Association, 2003, 1(1): 68-123.

[15] YI B, LI Z F, VIENS F G, et al. Robust optimal control for an insurer with reinsurance and investment under Heston's stochastic volatility model[J]. Insurance: Mathematics and Economics, 2013, 53(3): 601-614.

[16] 吴津津. 背景风险下DC型养老金的鲁棒最优投资策略[D]. 芜湖：安徽工程大学，2020.
WU J J. Robust optimal investment strategy of DC pension under background risk[D]. Wuhu: Anhui Polytechnic University, 2020. (in Chinese)

[17] 王倩，王沛祺，荣喜民. 考虑模型不确定的TBP养老金的鲁棒最优投资策略研究[J]. 经济数学，2020，37(3)：55-66.
WANG Q, WANG P Q, RONG X M. Robust optimal investment strategy under TBP pension model with model uncertainty[J]. Journal of Quantitative Economics, 2020, 37(3): 55-66. (in Chinese)

责任编辑：张自强