Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 310 (2016) 388–405
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

# An optimally accurate discrete regularization for second order timestepping methods for Navier–Stokes equations

Nan Jiang[a], Muhammad Mohebujjaman[b], Leo G. Rebholz[b,*], Catalin Trenchea[c]

[a] *Department of Scientific Computing, Florida State University, Tallahassee, FL 32312 United States*
[b] *Department of Mathematical Sciences, Clemson University, Clemson SC 29634, United States*
[c] *Department of Mathematics, 301 Thackeray Hall, University of Pittsburgh, Pittsburgh, PA 15260, United States*

Received 2 March 2016; received in revised form 13 July 2016; accepted 14 July 2016
Available online 26 July 2016

**Abstract**

We propose a new, optimally accurate numerical regularization/stabilization for (a family of) second order timestepping methods for the Navier–Stokes equations (NSE). The method combines a linear treatment of the advection term, together with stabilization terms that are proportional to discrete curvature of the solutions in both velocity and pressure. We rigorously prove that the entire new family of methods are unconditionally stable and $O(\Delta t^2)$ accurate. The idea of 'curvature stabilization' is new to CFD and is intended as an improvement over the commonly used 'speed stabilization', which is only first order accurate in time and can have an adverse effect on important flow quantities such as drag coefficients. Numerical examples verify the predicted convergence rate and show the stabilization term clearly improves the stability and accuracy of the tested flows.
© 2016 Elsevier B.V. All rights reserved.

*MSC:* 76D05; 65L20; 65M12

*Keywords:* Navier–Stokes; Unconditional stability; IMEX methods; Second order convergence; Crank–Nicolson; BDF2

## 1. Introduction

We consider optimally accurate stabilizations for second order time-stepping methods for the Navier–Stokes equations (NSE) on a bounded domain $\Omega \subseteq \mathbb{R}^d$, $d = 2$ or 3:

$$\begin{aligned} \mathbf{u}_t + \mathbf{u}\cdot\nabla\mathbf{u} - \nu\triangle\mathbf{u} + \nabla p = \mathbf{f}, \ \text{for } x \in \Omega, 0 < t \le T, \\ \nabla\cdot\mathbf{u} = 0, \text{ for } x \in \Omega, 0 < t \le T, \\ \mathbf{u} = 0, \text{ on } \partial\Omega, \text{ for } 0 < t \le T, \\ \mathbf{u}(x,0) = \mathbf{u}_0(x), \text{ for } x \in \Omega, \end{aligned} \tag{1.1}$$

with the pressure satisfying the usual zero-mean normalization.

* Corresponding author.
*E-mail addresses:* njiang@fsu.edu (N. Jiang), mmohebu@g.clemson.edu (M. Mohebujjaman), rebholz@clemson.edu (L.G. Rebholz), trenchea@pitt.edu (C. Trenchea).

http://dx.doi.org/10.1016/j.cma.2016.07.017
0045-7825/© 2016 Elsevier B.V. All rights reserved.

Developing efficient, accurate, and robust numerical methods for solving the NSE remains a great challenge in Computational Fluid Dynamics (CFD). For time-stepping methods, common approaches combine linearizations at each time step with stabilizations/regularizations that damp oscillations and unstable modes. An important linearization method is CNLE (Crank–Nicolson with linear extrapolation), proposed by Baker [1], which is comparable in stability and accuracy with the more expensive, fully implicit Crank–Nicolson (CN) method, [2–4]. However, while a nonlinear solver for CN requires several linear solves at each time step, CNLE requires just one. A similar linearization exists for BDF2 timestepping (called BDF2LE). Herein, we will consider a new, optimally accurate stabilization to be used with CNLE, BDF2LE, and the family of methods 'in between' them. Recall this family of methods (without stabilization) is given by

$$\frac{(\theta+\frac{1}{2})\mathbf{u}_{n+1}-2\theta\mathbf{u}_n+(\theta-\frac{1}{2})\mathbf{u}_{n-1}}{\triangle t}-\theta\nu\triangle\mathbf{u}_{n+1}-\nu(1-\theta)\triangle\mathbf{u}_n$$
$$+((\theta+1)\mathbf{u}_n-\theta\mathbf{u}_{n-1})\cdot\nabla(\theta\mathbf{u}_{n+1}+(1-\theta)\mathbf{u}_n)+\theta\nabla p_{n+1}+(1-\theta)\nabla p_n=\mathbf{f}_{n+\theta}, \tag{1.2}$$
$$\nabla\cdot\mathbf{u}_{n+1}=0, \tag{1.3}$$

where $\theta\in\left[\frac{1}{2},1\right]$. If $\theta=1$, BDF2LE is recovered, and if $\theta=\frac{1}{2}$, then CNLE is recovered. For any other $\theta\in(\frac{1}{2},0)$, a second order method is still recovered. Since CNLE exactly conserves energy, and BDF2LE numerically dissipates it, the parameter $\theta$ can be used to control the dissipation.

A successful stabilization method to be used with (1.2) must be able to damp the instabilities that frequently arise in NSE simulations, but without over-smoothing or removing important flow structures, i.e. without hurting accuracy. A common approach for these timestepping methods is to add $-\alpha\Delta\mathbf{u}_{n+1}$ to the left hand side, and $-\alpha\Delta\mathbf{u}_n$ to the right hand side, where $\alpha$ is a tuning parameter generally taken to be on the order of the meshwidth $h$. Such a stabilization has been used in methods for Navier–Stokes (see [5,6] and references therein), and is related in principle to techniques used in turbulence modeling [7,8], ocean modeling [9], and also to the discretization of the 'Voigt term' in a turbulence model recently studied by Titi and others, e.g. [10,11]. As shown in these works, this stabilization can be effective for several different types of flows, and also can improve conditioning of linear systems by increasing the coefficient of the stiffness matrix, e.g. in the case of BDF2LE, from $\nu$ to $\nu+\alpha$. However, as shown in the analysis of [5], this technique is $O(\alpha\Delta t)$ accurate, and thus can potentially be a dominant error source in second order timestepping methods if the usual choice of $\alpha=O(h)$ is made. If one instead takes $\alpha=O(\Delta t)$, this creates a need for a careful retuning of $\alpha$ each time the time step size is changed, which could make its use with adaptive time-stepping very difficult.

The purpose of this paper is to introduce and analyze a new stabilization for time-stepping methods of the form (1.2), that can sufficiently stabilize, but is $O(\Delta t^2)$ accurate, which is optimally accurate for second order timestepping methods. The design of the stabilization is inspired by the idea of stabilizing 'curvature' $(\mathbf{u}_{n+1}-2\mathbf{u}_n+\mathbf{u}_{n-1})$, instead of stabilizing 'speed' $(\mathbf{u}_{n+1}-\mathbf{u}_n)$, which is done by the stabilization discussed above. Not only is curvature stabilization more accurate than speed stabilization (with respect to $\Delta t$), but in CFD it does not directly alter important flow quantities such as drag coefficients, as speed penalization does (see Section 5.3). To our knowledge, the idea of curvature stabilization is new to CFD, and was first introduced very recently [12,13] as a timestepping method for two particular classes of ODEs. We note that an interface stabilization term for a Stokes–Darcy system in the recent paper [14] could also be considered to be in the same spirit, although their interpretation was somewhat different, and their error analysis led to a similar second order curvature-type term. In the numerical weather and climate prediction models, the Robert–Asselin time filter and its refinements (the Robert–Asselin–Williams [15], the higher-order Robert–Asselin time filter [16], etc.) have a similar stabilizing 'curvature' effect (see e.g. [17,18]).

The new family of second-order, unconditionally stable, IMEX time-stepping methods we propose are given by:

$$\frac{(\theta+\frac{1}{2})\mathbf{u}_{n+1}-2\theta\mathbf{u}_n+(\theta-\frac{1}{2})\mathbf{u}_{n-1}}{\triangle t}-\theta(\nu+\epsilon)\triangle\mathbf{u}_{n+1}-(\nu-\theta(\nu+2\epsilon))\triangle\mathbf{u}_n-\theta\epsilon\triangle\mathbf{u}_{n-1}$$
$$+((\theta+1)\mathbf{u}_n-\theta\mathbf{u}_{n-1})\cdot\nabla\left(\theta\frac{\nu+\epsilon}{\nu}\mathbf{u}_{n+1}+\left(1-\theta\frac{\nu+2\epsilon}{\nu}\right)\mathbf{u}_n+\theta\frac{\epsilon}{\nu}\mathbf{u}_{n-1}\right)$$
$$+\theta\frac{\nu+\epsilon}{\nu}\nabla p_{n+1}+\left(1-\theta\frac{\nu+2\epsilon}{\nu}\right)\nabla p_n+\theta\frac{\epsilon}{\nu}\nabla p_{n-1}=\mathbf{f}_{n+\theta}, \tag{1.4}$$
$$\nabla\cdot\mathbf{u}_{n+1}=0,$$

Download English Version:

https://daneshyari.com/en/article/497588

Download Persian Version:

https://daneshyari.com/article/497588

Daneshyari.com