A TIME-ACCURATE, ADAPTIVE DISCRETIZATION FOR FLUID FLOW PROBLEMS

VICTOR DECARIA, WILLIAM LAYTON, AND HAIYUN ZHAO

Abstract. This report presents a low computational and cognitive complexity, stable, time accurate and adaptive method for the Navier-Stokes equations. The improved method requires a minimally intrusive modification to an existing program based on the fully implicit / backward Euler time discretization, does not add to the computational complexity, and is conceptually simple. The backward Euler approximation is simply post-processed with a two-step, linear time filter. The time filter additionally removes the overdamping of Backward Euler while remaining unconditionally energy stable, proven herein. Even for constant stepsizes, the method does not reduce to a standard / named time stepping method but is related to a known 2-parameter family of A-stable, two step, second order methods. Numerical tests confirm the predicted convergence rates and the improved predictions of flow quantities such as drag and lift.

Key words. Navier-Stokes, backward Euler, time filter, time discretization, finite element method.

1. Introduction

The backward Euler time discretization is often used for complex, viscous flows due to its stability, rapid convergence to steady state solutions and simplicity to implement. However, it has poor time transient flow accuracy, [17], and can fail by overdamping a solution's dynamic behavior. For ODEs, adding a time filter to backward Euler, as in (1.3) below, yields two, embedded, A-stable approximations of first and second order accuracy, [20]. This report develops this idea into an adaptive time-step and adaptive order method for time accurate fluid flow simulation and gives an analysis of the resulting methods properties for constant time-steps. For constant time-steps, the resulting Algorithm 1.1 below involves adding only 1 extra line to a backward Euler code. The added filter step increases accuracy and adds negligible additional computational complexity, see Figure 1a and Figure 1b. Further, both time adaptivity and order adaptivity, presented in Section 2 and tested in Section 6, are easily implemented in a constant time step backward Euler code with $\mathcal{O}(20)$ added lines. Thus, algorithms herein have two main features. First, they can be implemented in a legacy code based on backward Euler without modifying the legacy components. Second, both time step and method order can easily be adapted due to the embedded structure of the method. The variable step, variable order (VSVO) method is presented in Section 2 and tested in Section 6.2.

Even for constant time-steps and constant order, the method herein does not reduce to a standard / named method. Algorithm 1.1 with Option B is (for constant order and time-step) equivalent to a member of the known, 2 parameter family of second order, 2-step, A-stable one leg methods (OLMs), see Algorithm 3.2, Section 3. Stability and velocity convergence of the (constant time step) general second

order, two-step, A-stable method for the Navier-Stokes equations was proven already in [15], see equation (3.20) p. 185, and has been elaborated thereafter, e.g., [23]. Our *velocity* stability and error analysis, while necessary for completeness, parallels this previous work and is thus collected in Appendix A. On the other hand, Algorithm 1.1 with Option A does *not* fit within a general theory even for constant stepsize, and produces more accurate pressure approximations.

We begin by presenting the simplest, constant stepsize case to fix ideas. Consider the time dependent incompressible Navier-Stokes (NS) equations:

$$u_t + u \cdot \nabla u - \nu \Delta u + \nabla p = f, \text{ and } \nabla \cdot u = 0 \text{ in } \Omega,$$

(1)
$$u = 0 \text{ on } \partial\Omega, \text{ and } \int_\Omega p \, dx = 0,$$

$$u(x,0) = u_0(x) \text{ in } \Omega.$$

Here, $\Omega \subset \mathbb{R}^d (d=2,3)$ is a bounded polyhedral domain; $u : \Omega \times [0,T] \to \mathbb{R}^d$ is the fluid velocity; $p : \Omega \times (0,T] \to \mathbb{R}$ is the fluid pressure. The body force $f(x,t)$ is known, and ν is the kinematic viscosity of the fluid.

Suppressing the spacial discretization, the method calculates an intermediate velocity $\hat{u}^{n+1}$ using the backward Euler / fully implicit method. Time filters (requiring only two additional lines of code and not affecting the BE calculation) are applied to produce u^{n+1} and p^{n+1} follows:

Algorithm 1.1 (Constant $\triangle t$ BE plus time filter). *With $u^* = \hat{u}^{n+1}$ (Implicit) or $u^* = 2u^n - u^{n-1}$ (Linearly-Implicit), Step 1: (Backward Euler)*

(2)
$$\frac{\hat{u}^{n+1} - u^n}{\Delta t} + u^* \cdot \nabla \hat{u}^{n+1} - \nu \Delta \hat{u}^{n+1} + \nabla \hat{p}^{n+1} = f(t^{n+1}),$$
$$\nabla \cdot \hat{u}^{n+1} = 0,$$

Step 2: (Time Filter for velocity and pressure)

(3)
$$u^{n+1} = \hat{u}^{n+1} - \frac{1}{3}(\hat{u}^{n+1} - 2u^n + u^{n-1})$$

Option A: (No pressure filter)

$$p^{n+1} = \hat{p}^{n+1}.$$

Option B:

$$p^{n+1} = \hat{p}^{n+1} - \frac{1}{3}(\hat{p}^{n+1} - 2p^n + p^{n-1})$$

Algorithm 1.1A means Option A is used, and Algorithm 1.1B means Option B is used.

Its implementation in a backward Euler code does not require additional function evaluations or solves, only a minor increase in floating point operations. Figure 1a presents a runtime comparison with and without the filter step. It is apparent that the added computational complexity of Step 2 is negligible. However, adding the time filter step has a profound impact on solution quality, see Figure 1b.

Herein, we give a velocity stability and error analysis for constant timestep in Appendix A. Since (eliminating the intermediate step) the constant time-step method is equivalent to an A-stable, second order, two step method, its velocity analysis has only minor deviations from the analysis in [15] and [23]. We also give an analysis of the unfiltered pressure error, which does not have a parallel in [15] or [23]. The
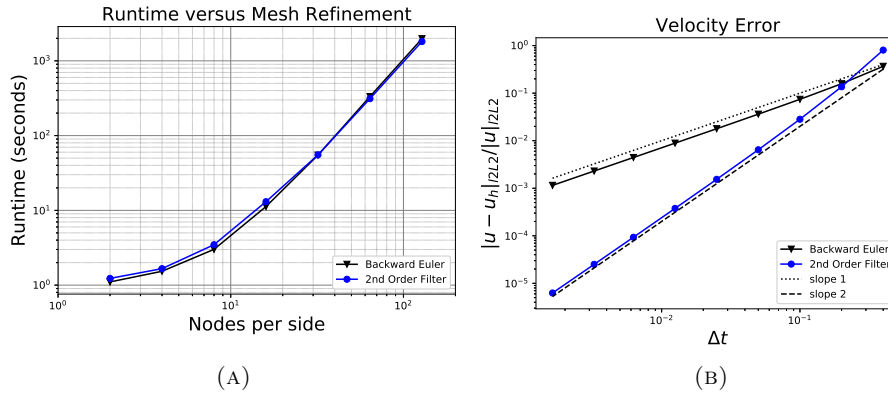
FIGURE 1. The time filter does not add to the computational complexity (Figure 1a), yet increases the method to second order (Figure 1b).

predicted (optimal) convergence rates are confirmed in numerical tests in Section 6. We prove the pressure approximation is stable and second order accurate provided only the velocity is filtered. The predicted second order pressure convergence, with or without filtering the pressure, is also confirmed in our tests, Figure 2.

The rest of the paper is organized as follow. In Section 2, we give the full, self-adaptive VSVO algorithm for a general initial value problem. Section 3 introduces some important mathematical notations and preliminaries necessary and analyze the method for the Navier-Stokes equations. In Section 4, we prove unconditional, nonlinear energy stability in Theorem 7. We analyze consistency error in Section 4.1. In Appendix A.2, we prove $\mathcal{O}(\Delta t^2)$ convergence for velocity, Theorem 10. The proof of the stability of the pressure is in Theorem 12 in Section 5.1. We prove second order accuracy for pressure in Section 5.2. Numerical tests are given in Section 6 to validate the theoretical predictions.

1.1. Related work. Time filters are primarily used to stabilize leapfrog time discretizations of weather models; see [30], [3], [34]. In [20] it was shown that the time filter used herein increases accuracy to second order, preserves A-stability, anti-diffuses the backward Euler approximation and yields an error estimator useful for time adaptivity. The analysis in [20] is an application of classical numerical ODE theory and does not extend to the Navier-Stokes equations. For the constant time step case, our analysis is based on eliminating the intermediate approximation $\hat{u}^{n+1}$ and reducing the method to an equivalent two step, OLM (a twin of a linear multistep method). The velocity stability and convergence of the general A-stable OLM was analyzed for the NSE (semi-implicit, constant time step and without space discretization) in [15]. Thus, the constant time step, discrete *velocity* results herein follow from these results. There is considerable previous work on analysis of multistep time discretizations of various PDEs, e.g. Crouzeix and Raviart [8]. Baker, Dougalis, and Karakashian [4] gave a long-time error analysis of the BDF methods for the NSE under a small data condition. (We stress that the method herein is *not a BDF method*.) The analysis of the method in Girault and Raviart [15] was extended to include spacial discretizations in [23]. The work in [23] also

shows how to choose those parameters to improve accuracy in higher Reynolds number flows - a significant contribution by itself. Other interesting extensions include the work of Gevici [14], Emmrich [10], [11], Jiang [22], Ravindran [29] and [25].

2. The Adaptive VSVO Method

Section 6.2 tests both the constant time step method and the method with adaptive step and adaptive order. This section will present the algorithmic details of adapting both the order and time step based on estimates of local truncation errors based on established methods [18]. The constant time step Algorithm 1.1 involves adding one (Option A) or two (Option B) lines to a backward Euler FEM code. The full self adaptive VSVO Algorithm 2.1 below adds $\mathcal{O}(20)$ lines. We first give the method for the initial value problem

$$y\,'(t) = f(t, y(t)), \text{ for } t > 0 \text{ and } y(0) = y_0.$$

Denote the nth time step size by Δt_n. Let $t^{n+1} = t^n + \Delta t_n$ and y^n an approximation to $y(t_n)$. The choice of filtering weights depend on $\omega_n \coloneqq \Delta t_n / \Delta t_{n-1}$, Step 2 below. TOL is the user supplied tolerance on the allowable error per step.

Algorithm 2.1 (Variable Stepsize, Variable Order 1 and 2 (VSVO-12)).
Step 1 : Backward Euler

$$\frac{y_{(1)}^{n+1} - y^n}{\Delta t_n} = f(t_{n+1}, y_{(1)}^{n+1})$$

Step 2 : Time Filter

$$y_{(2)}^{n+1} = y_{(1)}^{n+1} - \frac{\omega_n}{2\omega_n + 1}\left(y_{(1)}^{n+1} - (1 + \omega_n)y^n + \omega_n y^{n-1}\right)$$

Step 3 : Estimate error in $y_{(1)}^{n+1}$ and $y_{(2)}^{n+1}$.

$$EST_1 = y_{(2)}^{n+1} - y_{(1)}^{n+1}$$

$$EST_2 = \frac{\omega_{n-1}\omega_n(1 + \omega_n)}{1 + 2\omega_n + \omega_{n-1}\left(1 + 4\omega_n + 3\omega_n^2\right)}\left(y_{(2)}^{n+1}\right.$$
$$-\frac{(1 + \omega_n)(1 + \omega_{n-1}(1 + \omega_n))}{1 + \omega_{n-1}}y^n + \omega_n(1 + \omega_{n-1}(1 + \omega_n))y^{n-1}$$
$$\left.-\frac{\omega_{n-1}^2\omega_n(1 + \omega_n)}{1 + \omega_{n-1}}y^{n-2}\right).$$

Step 4 : Check if tolerance is satisfied.
If $\|EST_1\| < TOL$ or $\|EST_2\| < TOL$, at least one approximation is acceptable. Go to Step 5a. Otherwise, the step is rejected. Go to Step 5b.
Step 5a : At least one approximation is accepted. Pick an order and stepsize to proceed.
If both approximations are acceptable, set

$$\Delta t^{(1)} = 0.9\Delta t_n\left(\frac{TOL}{\|EST_1\|}\right)^{\frac{1}{2}}, \qquad \Delta t^{(2)} = 0.9\Delta t_n\left(\frac{TOL}{\|EST_2\|}\right)^{\frac{1}{3}}.$$

Set

$$i = \arg\max_{i \in \{1,2\}} \Delta t^{(i)}, \qquad \Delta t_{n+1} = \Delta t^{(i)}, \qquad t^{n+2} = t^{n+1} + \Delta t_{n+1}, \qquad y^{n+1} = y_{(i)}^{n+1}.$$

If only $y^{(1)}$ (resp. $y^{(2)}$) satisfies TOL, set $\Delta t_{n+1} = \Delta t^{(1)}$ (resp. $\Delta t^{(2)}$), and $y^{n+1} = y^{n+1}_{(1)}$ (resp. $y^{n+1}_{(2)}$). Proceed to Step 1 to calculate y^{n+2}.

Step 5b : Neither approximations satisfy TOL.

Set

$$\Delta t^{(1)} = 0.7 \Delta t_n \left(\frac{TOL}{\|EST_1\|} \right)^{\frac{1}{2}}, \qquad \Delta t^{(2)} = 0.7 \Delta t_n \left(\frac{TOL}{\|EST_2\|} \right)^{\frac{1}{3}}.$$

Set

$$i = \arg\max_{i \in \{1,2\}} \Delta t^{(i)}, \qquad \Delta t_n = \Delta t^{(i)}, \qquad t^{n+1} = t^n + \Delta t_n$$

Return to Step 1 to try again.

For clarity, we have not mentioned several standard features such as setting a maximum and minimum timestep, the maximum or minimum stepsize ratio, etc.

The implementation above computes an estimation of the local errors in Step 3. EST_1 provides an estimation for the local error of the first order approximation $y^{(1)}_{n+1}$ since $y^{(2)}_{n+1}$ is a second order approximation. For a justification of EST_2, see Appendix B.

Standard formulas, see e.g. [19], are used to pick the next stepsize in Steps 5a and 5b. Based on the previous Δt and the current error estimator, the formula estimates the largest next stepsize that can be taken by the method such that the tolerance will still be satisfied. Out of the approximations that satisfy the tolerance, the approximation which yielded the largest estimated Δt is chosen to advance the solution.

The numbers 0.9 in Step 5a and 0.7 in Step 5b are commonly used safety factors to make the next approximation more likely to be accepted since the exact optimal Δt is unknowable.

One more line is needed for linearly implicit methods. For linearly implicit methods the point of linearization must also have $\mathcal{O}(\Delta t^2)$ accuracy. For example, with $u^* = u^n$

(4)
$$\frac{u^{n+1} - u^n}{\Delta t_n} + u^* \cdot \nabla u^{n+1} + \frac{1}{2} (\nabla \cdot u^*) u^{n+1} + \nabla p^{n+1} - \nu \Delta u^{n+1} = f^{n+1} \ \& \ \nabla \cdot u^{n+1} = 0$$

is a common first order linearly implicit method. The required modification in the BE step to ensure second order accuracy after the filter is to shift the point of linearization from $u^* = u^n$ to

$$u^* = \left(1 + \frac{\Delta t_n}{\Delta t_{n-1}} \right) u^n - \frac{\Delta t_n}{\Delta t_{n-1}} u^{n-1} = (1 + \omega_n) u^n - \omega_n u^{n-1}.$$

Other simplifications. The algorithm can be simplified if only the time-step is adapted (not order adaptive). It can be further simplified using *extrapolation* where the second order approximation is adapted based on EST_1 (pessimistic for the second order approximation).

3. Notations and preliminaries

We introduce some notations and inequalities which will be used in later sections. $(\cdot, \cdot), \|\cdot\|$ denotes the $L^2(\Omega)$ inner product and norm. C will denote a generic, finite constant depending possibly on T, Ω and f. The velocity space X and pressure

space Q are defined

$$X := H_0^1(\Omega)^d = \{v \in H^1(\Omega)^d : v|_{\partial\Omega} = 0\},$$

$$Q := L_0^2(\Omega)^d = \{q \in L^2(\Omega) : \int_\Omega q = 0\}.$$

The divergence free space V is given by

$$V := \{v \in X : (\nabla \cdot v, q) = 0 \quad \forall q \in Q)\}.$$

For measurable $v : [0, T] \to X$, define for, respectively, $1 \leq p < \infty$ and $p = \infty$

$$||v||_{L^p(0,T;X)} = \left(\int_0^T ||v(t)||_X^p dt\right)^{1/p} \quad \text{and} \quad ||v||_{L^\infty(0,T;X)} = ess \sup_{0 \leq t \leq T} ||v(t)||_X,$$

$$|||v|||_{p,k} = \left(\int_0^T ||v(t)||_{H^k(\Omega)}^p\right)^{1/p} \quad \text{and} \quad |||v|||_{\infty,k} = ess \sup_{0 \leq t \leq T} ||v(t)||_{H^k(\Omega)}.$$

We define the skew-symmetrized nonlinear form:

$$B(u,v) := u \cdot \nabla v + \frac{1}{2}(\nabla \cdot u)v, \quad \forall\, u, v, w \in X,$$

$$b(u,v,w) := (B(u,v), w).$$

Lemma 1. *There exists $C > 0$ such that*

$$b(u,v,w) \leq C\|\nabla u\|\|\nabla v\|\|\nabla w\|, \qquad \forall\, u, v, w \in X$$

$$b(u,v,w) \leq C\|u\|\|v\|_2\|\nabla w\| \quad \forall u, w \in X, v \in X \cap H^2(\Omega).$$

Proof. See Lemma 2.1 on p. 12 of [32]. $\qquad\qquad\qquad\qquad\qquad\qquad\square$

We use the following discrete Gronwall inequality found in [21, Lemma 5.1].

Lemma 2 (Discrete Gronwall Inequality). *Let Δt, H, a_n, b_n, c_n, d_n (for integers $n \geq 0$) be non-negative numbers such that*

$$(5) \qquad a_N + \Delta t \sum_{n=0}^N b_n \leq \Delta t \sum_{n=0}^N d_n a_n + \Delta t \sum_{n=0}^N c_n + H, \quad \forall\, N \geq 0$$

Suppose $\Delta t d_n < 1 \,\forall n$, then,

$$(6) \qquad a_N + \Delta t \sum_{n=0}^N b_n \leq \exp\left(\Delta t \sum_{n=0}^N \frac{1}{1 - \Delta t d_n}\right)\left(\Delta t \sum_{n=0}^N c_n + H\right), \quad \forall\, N \geq 0$$

Multiplying (1) by test functions $(v, q) \in (X, Q)$ and integrating by parts gives

$$(7) \quad (u_t, v) + b(u, u, v) + \nu(\nabla u, \nabla v) - (p, \nabla \cdot v) + (\nabla \cdot u, q) = (f, v), \quad (\nabla \cdot u, q) = 0.$$

To discretize the above system in space, we choose conforming finite element spaces for velocity $X_h \subset X$ and pressure $Q_h \subset Q$ satisfying the discrete inf-sup condition and the following approximation properties:

$$(8) \qquad\qquad \inf_{q_h \in Q_h} \sup_{v_h \in X_h} \frac{(q_h, \nabla \cdot v_h)}{\|q_h\|\|\nabla v_h\|} \geq \beta > 0.$$

We further assume that for each $u \in X \cap H^{k+1}(\Omega)^d$, and $p \in Q \cap H^{s+1}(\Omega)$ there exists $v_h \in X_h$ and $q_h \in Q_h$ such that

$$\|u - v_h\| \le Ch^{k+1}\|u\|_{k+1},$$
(9)
$$\|u - v_h\|_1 \le Ch^k\|u\|_{k+1},$$
$$\|p - q_h\| \le Ch^{s+1}\|p\|_{s+1}.$$

h denotes the maximum triangle diameter. Examples of finite element spaces satisfying these conditions are the MINI [2] and Taylor-Hood [33] elements. The discretely divergence free subspace $V_h \in X_h$ is defined

$$V_h := \{v_h \in X_h : (\nabla \cdot v_h, q_h) = 0 \quad \forall q_h \in Q_h\}.$$

The following Lemma establishes that if the discrete inf-sup condition (8) and (9) hold, then V_h also has good approximation properties. It is shown in the proof of Theorem 1.1 in Chapter II of [16].

Lemma 3. *Suppose the pair* (X_h, Q_h) *satisfies* (8). *Then for all* $u \in V$,

$$\inf_{v_h \in V_h} \|\nabla (u - v_h)\| \le \inf_{v_h \in X_h} C(\beta)\|\nabla (u - v_h)\|.$$

The dual norms of X_h and V_h are

$$\|w\|_{X_h^*} := \sup_{v_h \in X_h} \frac{(w, v_h)}{\|\nabla v_h\|}, \|w\|_{V_h^*} := \sup_{v_h \in V_h} \frac{(w, v_h)}{\|\nabla v_h\|}.$$

The following Lemma from Galvin [13, p. 243] establishes the equivalence of these norms on V_h.

Lemma 4. *Suppose the discrete inf-sup condition holds, let* $w \in V_h$, *then there exists* $C > 0$, *independent of* h, *such that*

$$C\|w\|_{X_h^*} \le \|w\|_{V_h^*} \le \|w\|_{X_h^*}.$$

Lemma 4 is used to derive pressure error estimates with a technique shown in Fiordilino [12]. We will use the following, easily proven, algebraic identity.

Lemma 5. *The following identity holds.*

(10)
$$\left(\frac{3}{2}a - 2b + \frac{1}{2}c\right)\left(\frac{3}{2}a - b + \frac{1}{2}c\right) =$$
$$\left(\frac{a^2}{4} + \frac{(2a - b)^2}{4} + \frac{(a - b)^2}{4}\right) - \left(\frac{b^2}{4} + \frac{(2b - c)^2}{4} + \frac{(b - c)^2}{4}\right) + \frac{3}{4}(a - 2b + c)^2.$$

With the notation in place, we state the fully discrete method.

Algorithm 3.1 (Fully Discrete Method). *Given* $u_h^{n-1}, u_h^n \in X_h$ *(and if necessary, given* $p_h^{n-1}, p_h^n \in Q_h$), *find* $(\hat{u}_h^{n+1}, \hat{p}^{n+1}) \in (X_h, Q_h)$ *satisfying*

$$\left(\frac{\hat{u}_h^{n+1} - u_h^n}{\Delta t_n}, v_h\right) + b(\hat{u}_h^{n+1}, \hat{u}_h^{n+1}, v_h) + \nu(\nabla \hat{u}_h^{n+1}, \nabla v_h)$$
(11)
$$-(\hat{p}_h^{n+1}, \nabla \cdot v_h) = (f(t^{n+1}), v_h),$$
$$(\nabla \cdot \hat{u}^{n+1}, q_h) = 0.$$

for all $(v_h, q_h) \in (X_h, Q_h)$. *Then compute*

$$u_h^{n+1} = \hat{u}_h^{n+1} - \frac{\omega_n}{2\omega_n + 1}\left(\hat{u}_h^{n+1} - (1 + \omega_n)u_h^n + \omega_n u_h^{n-1}\right).$$

Option A: (No pressure filter)

$$p_h^{n+1} = \hat{p}_h^{n+1}.$$

Option B:

$$p_h^{n+1} = \hat{p}_h^{n+1} - \frac{\omega_n}{2\omega_n + 1}\left(\hat{p}_h^{n+1} - (1+\omega_n)p_h^n + \omega_n p_h^{n-1}\right).$$

The constant time-step stability and error analysis works with the following equivalent formulation of the method. We stress that what follows is not the preferred implementation since it only yields one approximation, while Algorithm 3.1 gives the embedded approximations $\hat{u}_h^{n+1}$ and u_h^{n+1} and an error estimator.

Algorithm 3.2 (Constant time-step, equivalent method). *Assume the time-step is constant. Given* (u_h^n, p_h^n) *and* (u_h^{n-1}, p_h^{n-1})*, find* (u_h^{n+1}, p_h^{n+1}) *such that for all* $(v_h, q_h) \in (X_h, Q_h)$,
Option A

(12)
$$\left(\frac{\frac{3}{2}u_h^{n+1} - 2u_h^n + \frac{1}{2}u_h^{n-1}}{\Delta t}, v_h\right) + b\left(\frac{3}{2}u_h^{n+1} - u_h^n + \frac{1}{2}u_h^{n-1}, \frac{3}{2}u_h^{n+1} - u_h^n + \frac{1}{2}u_h^{n-1}, v_h\right)$$
$$+\nu\left(\nabla\left(\frac{3}{2}u_h^{n+1} - u_h^n + \frac{1}{2}u_h^{n-1}\right), \nabla v_h\right) - \left(\mathbf{p_h^{n+1}}, \boldsymbol{\nabla} \cdot \mathbf{v_h}\right)$$
$$= \left(f^{n+1}, v_h\right),$$
$$\left(\nabla \cdot \left(\frac{3}{2}u_h^{n+1} - u_h^n + \frac{1}{2}u_h^{n-1}\right), q_h\right) = 0,$$

or Option B

(13)
$$\left(\frac{\frac{3}{2}u_h^{n+1} - 2u_h^n + \frac{1}{2}u_h^{n-1}}{\Delta t}, v_h\right) + b\left(\frac{3}{2}u_h^{n+1} - u_h^n + \frac{1}{2}u_h^{n-1}, \frac{3}{2}u_h^{n+1} - u_h^n + \frac{1}{2}u_h^{n-1}, v_h\right)$$
$$+\nu\left(\nabla\left(\frac{3}{2}u_h^{n+1} - u_h^n + \frac{1}{2}u_h^{n-1}\right), \nabla v_h\right) - \left(\mathbf{\frac{3}{2}p_h^{n+1} - p_h^n + \frac{1}{2}p_h^{n-1}}, \boldsymbol{\nabla} \cdot \mathbf{v_h}\right)$$
$$= \left(f^{n+1}, v_h\right),$$
$$\left(\nabla \cdot \left(\frac{3}{2}u_h^{n+1} - u_h^n + \frac{1}{2}u_h^{n-1}\right), q_h\right) = 0.$$

The pressure is highlighted in bold, and is the only difference between the two above equations. The time difference term of the above equivalent method is that of BDF2 but the remainder is different. This is *not* the standard BDF2 method.

Proposition 6. *Algorithm 3.1A (respectively B) is equivalent Algorithm 3.2A (respectively B).*

Proof. We will just prove the case for Option A since the other case is similar. Let (u_h^{n+1}, p_h^{n+1}) be the solution to Algorithm 3.1. By linearity of the time filter, $(u_h^{n+1}, p_h^{n+1}) \in (X_h, Q_h)$. We can write $\hat{u}_h^{n+1}$ in terms of u_h^{n+1}, u_h^n, and u_h^{n-1} as $\hat{u}^{n+1} = \frac{3}{2}u^{n+1} - u^n + \frac{1}{2}u^{n-1}$. Substitute this into (11). Then (u_h^{n+1}, p_h^{n+1}) satisfies equation (12).

These steps can be reversed to show the converse. $\square$

We next define the discrete kinetic energy, viscous and numerical dissipation terms that arise naturally from a G-stability analysis of Algorithm 3.2, *regardless* of whether Option A or B is used. The (constant time-step) discrete kinetic energy, discrete viscous energy dissipation rate and the numerical energy dissipation rate of Algorithm 3.2 are

$$
\begin{array}{ll}
\text{discrete energy:} & \mathcal{E}^n = \frac{1}{4}\left[\|u^n\|^2 + \|2u^n - u^{n-1}\|^2 + \|u^n - u^{n-1}\|^2\right], \\
\text{viscous dissipation:} & \mathcal{D}^{n+1} = \Delta t\nu\|\nabla\left(\frac{3}{2}u^{n+1} - u^n + \frac{1}{2}u^{n-1}\right)\|^2, \\
\text{numerical dissipation:} & \mathcal{Z}^{n+1} = \frac{3}{4}\|u^{n+1} - 2u^n + u^{n-1}\|^2.
\end{array}
$$

Remark 1. *As $\Delta t \to 0$, $\mathcal{E}^n$ is consistent with the kinetic energy $\frac{1}{2}\|u\|^2$ and $\mathcal{D}^n$ is consistent with the instantaneous viscous dissipation $\nu\|\nabla u\|^2$. The numerical dissipation $\mathcal{Z}^{n+1} \approx \frac{3}{4}\Delta t^4\|u_{tt}(t^{n+1})\|^2$, is asymptotically smaller than the numerical dissipation of backward Euler, $\frac{1}{2}\Delta t^2\|u_t(t^{n+1})\|^2$.*

The method's kinetic energy differs from that of BDF2, which is (e.g. [26])

$$
\mathcal{E}^n_{BDF2} = \frac{1}{4}\left[\|u^n\|^2 + \|2u^n - u^{n-1}\|^2\right]
$$

due to the term $\|u^n - u^{n-1}\|^2$ in $\mathcal{E}^n$ which is a dispersive penalization of a discrete acceleration.

Define the interpolation and difference operators as follows

Definition 1. *The interpolation operator I and difference operator D are*

$$
I[w^{n+1}] = \frac{3}{2}w^{n+1} - w^n + \frac{1}{2}w^{n-1} \quad and \quad D[w^{n+1}] = \frac{3}{2}w^{n+1} - 2w^n + \frac{1}{2}w^{n-1}.
$$

Formally, $I[w(t^{n+1})] = w(t^{n+1}) + \mathcal{O}(\Delta t^2)$, and $\frac{D[w(t^{n+1})]}{\Delta t} = w_t(t^{n+1}) + \mathcal{O}(\Delta t^2)$. This will be made more precise in the consistency error analysis in Section 4.1.

4. Stability and Error Analysis

We prove stability and error analysis of the *constant time-step* method. The velocity proofs parallel ones in [15] and [23] and are collected in Appendix A. The pressure analysis is presented in Section 5.

Theorem 7. *Assume the stepsize is constant. The following equality holds.*

$$
\mathcal{E}^N + \sum_{n=1}^{N-1}\mathcal{D}^{n+1} + \sum_{n=1}^{N-1}\mathcal{Z}^{n+1} = \Delta t\sum_{n=1}^{N-1}(f, I[u_h^{n+1}]) + \mathcal{E}^1.
$$

Proof. In Algorithm 3.2, set $v_h = \Delta t I[u_h^{n+1}]$ and $q_h = p_h^{n+1}$ for Option A, or $q_h = I[p_h^{n+1}]$ for Option B, and add.

$$
(14) \qquad\qquad (D[u_h^{n+1}], I[u_h^{n+1}]) + \mathcal{D}^{n+1} = \Delta t(f, I[u_h^{n+1}]).
$$

By Lemma 5 and Definition 1,

$$
(D[u_h^{n+1}], I[u_h^{n+1}]) = \mathcal{E}^{n+1} - \mathcal{E}^n + \mathcal{Z}^{n+1}.
$$

Thus, (14) can be written

$$
\mathcal{E}^{n+1} - \mathcal{E}^n + \mathcal{D}^{n+1} + \mathcal{Z}^{n+1} = \Delta t(f(t^{n+1}), I[u_h^{n+1}]).
$$

Summing over n from 1 to $N-1$ yields the result. $\qquad\square$

This result is for the time stepping method applied to the Navier-Stokes equations. More generally, the constant time-step method of Algorithm 1.1 is G-Stable, a fact that follows from the equivalence of A and G-Stability [9]. We calculate the G matrix explicitly below.

Corollary 8. *Assume the time-step is constant. Backward Euler followed by the time filter is G-Stable with G matrix*

$$G = \begin{bmatrix} \frac{3}{2} & -\frac{3}{4} \\ -\frac{3}{4} & \frac{1}{2} \end{bmatrix}.$$

Proof. Simply check that

$$[u^n, u^{n-1}]G \begin{bmatrix} u^n \\ u^{n-1} \end{bmatrix} = \frac{1}{4}\Big[|u^n|^2 + |2u^n - u^{n-1}|^2 + |u^n - u^{n-1}|^2\Big].$$

$\square$

4.1. Consistency error. By manipulating (7), we derive the consistency error. The true solution to (7) satisfies

$$
\begin{aligned}
(15) \quad & \left(\frac{D[u(t^{n+1})]}{\Delta t}, v_h\right) + b\left(I[u(t^{n+1})], I[u(t^{n+1})], v_h\right) \\
& + \nu\left(\nabla I[u(t^{n+1})], \nabla v_h\right) - \left(p(t^{n+1}), \nabla \cdot v_h\right) \\
& = \left(\mathbf{f}^{n+1}, v_h\right) + \tau^{n+1}(u, p; v_h) \quad \forall v_h \in X_h.
\end{aligned}
$$

If Option A is used (pressure is unfiltered),

$$
\begin{aligned}
(16) \quad \tau^{n+1}(u, p; v_h) = \tau_A^{n+1}(u, p; v_h) := & \left(\frac{D[u(t^{n+1})]}{\Delta t} - u_t(t^{n+1}), v_h\right) \\
& + b\left(I[u(t^{n+1})], I[u(t^{n+1})], v_h\right) - b(u(t^{n+1}), u(t^{n+1}), v_h) \\
& + \nu\left(\nabla(I[u(t^{n+1})] - u(t^{n+1})), \nabla v_h\right).
\end{aligned}
$$

If Option B is used (pressure is filtered),

$$(17) \quad \tau^{n+1}(u, p; v_h) = \tau_A^{n+1}(u, p; v_h) - \left(I[p(t^{n+1})] - p(t^{n+1}), \nabla \cdot v_h\right).$$

Thus, filtering the pressure introduces a term that, while still second order, adds to the consistency error. We believe this is why Option A performs better in the numerical tests, Figure 2. Furthermore, Option B requires assuming additional regularity for convergence, see Theorem 10.

The terms in the consistency error are bounded in the following lemma.

Lemma 9 (Consistency). *For u, p sufficiently smooth, we have*

$$\left\|\frac{D[u(t^{n+1})]}{\Delta t} - u_t(t^{n+1})\right\|^2 \le \frac{6}{5}\Delta t^3 \int_{t^{n-1}}^{t^{n+1}} \|u_{ttt}\|^2 dt,$$

$$(18) \quad \left\|I[u(t^{n+1})] - u(t^{n+1})\right\|^2 \le \frac{4}{3}\Delta t^3 \int_{t^{n-1}}^{t^{n+1}} \|u_{tt}\|^2 dt,$$

$$(19) \quad \left\|I[p(t^{n+1})] - p(t^{n+1})\right\|^2 \le \frac{4}{3}\Delta t^3 \int_{t^{n-1}}^{t^{n+1}} \|p_{tt}\|^2 dt.$$

Proof. See Appendix A. $\square$

4.2. Error estimates for the velocity. Next, we analyze the convergence of Algorithm 3.2 and give an error estimate for the velocity. Let $t^n = n\Delta t$. Denote the errors $\mathbf{e}_u^n = u(t^n) - u_h^n$ and $e_p^n = p(t^n) - p_h^n$.

Theorem 10. *Assume that the true solution (u, p) satisfies the following regularity*
(20)
$$u \in L^\infty(0, T; (H^{k+1}\Omega))^d), \quad u_t \in L^2(0, T; (H^{k+1}\Omega))^d), \quad u_{tt} \in L^2(0, T; (H^1\Omega))^d),$$
$$u_{ttt} \in L^2(0, T; (L^2\Omega))^d), \qquad p \in L^2(0, T; (H^{s+1}(\Omega))^d).$$

Additionally for Option B, assume $p_{tt} \in L^2(0, T; (L^2(\Omega))^d$. For (u_h^{n+1}, p_h^{n+1}) satisfying (12), and for Δt sufficiently small, we have the following estimate

(21)
$$\|e_u^N\|^2 + \|2e_u^N - e_u^{N-1}\|^2 + \|e_u^N - e_u^{N-1}\|^2 + \sum_{n=1}^{N-1} 3\|e_u^{n+1} - 2e_u^n + e_u^{n-1}\|^2$$
$$+ \nu\Delta t \sum_{n=1}^{N-1} \|\nabla I[e_u^{n+1}]\|^2 \le C\Big(h^{2k} + h^{2s+2} + \Delta t^4\Big).$$

Proof. See Appendix A. $\qquad\qquad\square$

5. Pressure Stability and Convergence

5.1. Stability of Pressure. We introduce the following discrete norms

(22) $$\||\omega|\|_{\infty,k} := \max_{0 \le n \le T/\Delta t} \|\omega^n\|_k, \quad \||\omega|\|_{2,k} := \left(\sum_{n=0}^{T/\Delta t - 1} \Delta t\|\omega^n\|_k^2\right)^{1/2}.$$

In this section, we prove that the pressure approximation is stable in $l^1(0, T; L^2(\Omega))$. We first give a corollary of Theorem 7 asserting the stability of the velocity approximation.

Corollary 11. *Suppose $f \in L^2(0, T; H^{-1}(\Omega)^d)$, then the velocity approximation satisfies*

$$\mathcal{E}^N + \frac{1}{2}\sum_{n=1}^{N-1} \mathcal{D}^{n+1} + \sum_{n=1}^{N-1} \mathcal{Z}^{n+1} \le \frac{1}{2\nu}\||f|\|_{2,-1}^2 + \mathcal{E}^1.$$

Proof. Consider Theorem 7. Applying the Cauchy-Schwarz yields the inequality. $\qquad\qquad\square$

We now prove the stability of the filtered pressure.

Theorem 12. *Suppose Corollary 11 holds, then the pressure approximation satisfies*

(23)
$$\beta\Delta t \sum_{n=1}^{N-1} \|p_h^{n+1}\| \le C \qquad \text{for Option A,}$$
$$\beta\Delta t \sum_{n=1}^{N-1} \|I[p_h^{n+1}]\| \le C \quad \text{for Option B.}$$

Proof. We prove it for Option A, as the other case is similar. Isolating the discrete time derivative in (12), and restricting v_h to V_h yields

$$(24) \qquad \begin{aligned} \left(\frac{D[u_h^{n+1}]}{\Delta t}, v_h\right) &= -b\left(I[u_h^{n+1}], I[u_h^{n+1}], v_h\right) \\ &- \nu\left(\nabla I[u_h^{n+1}], \nabla v_h\right) + \left(f^{n+1}, v_h\right) \quad \forall v_h \in V_h. \end{aligned}$$

The terms on the right hand side of (24) can be bounded as follows,

$$(25) \qquad \begin{aligned} b\left(I[u_h^{n+1}], I[u_h^{n+1}], v_h\right) &\le C\|\nabla I[u_h^{n+1}]\|\|\nabla I[u_h^{n+1}]\|\|\nabla v_h\|, \\ -\nu\left(\nabla I[u_h^{n+1}], \nabla v_h\right) &\le \nu\|\nabla I[u_h^{n+1}]\|\|\nabla v_h\|, \\ \left(f^{n+1}, v_h\right) &\le \|f^{n+1}\|_{-1}\|\nabla v_h\|. \end{aligned}$$

In equation (24), we can use the above estimates in (25), divide both sides by $\|\nabla v_h\|$, and take the supremum over $v_h \in V_h$. This gives

$$(26) \qquad \left\|\frac{D[u_h^{n+1}]}{\Delta t}\right\|_{V_h^*} \le (C\|\nabla I[u_h^{n+1}]\| + \nu)\|\nabla I[u_h^{n+1}]\| + \|f^{n+1}\|_{-1}.$$

Lemma 4 implies

$$(27) \qquad \left\|\frac{D[u_h^{n+1}]}{\Delta t}\right\|_{X_h^*} \le C\Big[(\|\nabla I[u_h^{n+1}]\| + 1)\|\nabla I[u_h^{n+1}]\| + \|f^{n+1}\|_{-1}\Big].$$

Now consider Algorithm 3.2 again with $v_h \in X_h$. Isolating the pressure term in (12) and using the estimates from (25) yields

$$(28) \qquad \left(p_h^{n+1}, \nabla \cdot v_h\right) \le \left(\frac{D[u_h^{n+1}]}{\Delta t}, v_h\right)$$

$$+ C(\|\nabla I[u_h^{n+1}]\| + 1)\|\nabla I[u_h^{n+1}]\|\|\nabla v_h\| + \|f^{n+1}\|_{-1}\|\nabla v_h\|.$$

Divide both sides by $\|\nabla v_h\|$, take supremum over $v_h \in X_h$ and use the discrete inf-sup condition and the results in (28). Then,

$$(29) \qquad \beta\|p_h^{n+1}\| \le C\Big[(\|\nabla I[u_h^{n+1}]\| + 1)\|\nabla I[u_h^{n+1}]\| + \|f^{n+1}\|_{-1}\Big].$$

We then multiply by Δt, sum from $n = 1$ to $n = N-1$, and apply Cauchy-Schwarz on the right hand side,

$$(30) \qquad \beta\Delta t \sum_{n=1}^{N-1} \|p_h^{n+1}\| \le C\Delta t\Big[(\||\nabla I[u_h]|\|_{2,0} + 1)\||\nabla I[u_h]|\|_{2,0} + \|f\|_{2,-1}\Big].$$

Then using the result from velocity approximation, we get,

$$(31) \qquad \beta\Delta t \sum_{n=1}^{N-1} \|p_h^{n+1}\| \le C\Big[(\||f|\|_{2,-1} + 1)\||f|\|_{2,-1} + (\mathcal{E}^1 + 1)\mathcal{E}^1\Big].$$

$\square$

5.2. Error estimates for the pressure. We now prove convergence of the pressure approximation in $l^1(0, T; L^2(\Omega))$. Denote the pressure error as $e_p^n = p(t^n) - p_h^n$.

Theorem 13. *Let u, p satisfy the equation* (21)*. Let the assumption of regularity in Theorem* 10 *be satisfied. Then there exists a constant $C > 0$ such that*

(32)
$$\Delta t \beta \sum_{n=1}^{N-1} \|e_p^{n+1}\| \le C\Big(h^k + h^{s+1} + \Delta t^2\Big) \qquad \text{for Option A,}$$
$$\Delta t \beta \sum_{n=1}^{N-1} \|I[e_p^{n+1}]\| \le C\Big(h^k + h^{s+1} + \Delta t^2\Big) \quad \text{for Option B.}$$

Proof. Again, we only prove this for Option A since the other case requires only slight modification. Using the equations (A.4) and (A.5) yields

(33)
$$\left(\frac{D[\phi_h^{n+1}]}{\Delta t}, v_h\right) = -\left(\frac{D[\eta^{n+1}]}{\Delta t}, v_h\right) - b\left(I[e_u^{n+1}], I[u(t^{n+1})], v_h\right)$$
$$- b\left(I[u_h^{n+1}], I[e_u^{n+1}], v_h\right) - \nu\left(\nabla I[e_u^{n+1}], \nabla v_h\right)$$
$$+ \left(e_p^{n+1}, \nabla \cdot v_h\right) + \tau^{n+1}(u, p; v_h) \quad \forall v_h \in V_h.$$

We bound the six individual terms on the right hand side of (33), term by term as follows:

(34)
$$\left(\frac{D[\eta^{n+1}]}{\Delta t}, v_h\right) \le C\Delta t^{-\frac{1}{2}} \|\eta_t\|_{L^2(t^{n-1}, t^{n+1}; L^2(\Omega))} \|\nabla v_h\|,$$

(35)
$$- b\left(I[e_u^{n+1}], I[u(t^{n+1})], v_h\right) \le C\|\nabla I[e_u^{n+1}]\|\|\nabla I[u(t^{n+1})]\|\|\nabla v_h\|,$$

(36)
$$- b\left(I[u_h^{n+1}], I[e_u^{n+1}], v_h\right) \le C\|\nabla(I[u_h^{n+1}])\|\|\nabla I[e_u^{n+1}]\|\|\nabla v_h\|,$$

(37)
$$- \nu\left(\nabla I[e_u^{n+1}], \nabla v_h\right) \le \nu\|\nabla I[e_u^{n+1}]\|\|\nabla v_h\|,$$

(38)
$$\left(p(t^{n+1}) - \lambda_h^{n+1}, \nabla \cdot v_h\right) \le C\|p(t^{n+1}) - \lambda_h^{n+1}\|\|\nabla v_h\|,$$

(39)
$$\tau^{n+1}(u, p; v_h) \le C\Delta t^{\frac{3}{2}} \Big(\|u_{ttt}\|_{L^2(t^{n-1}, t^{n+1}; L^2(\Omega))} + \|\nabla u_{tt}\|_{L^2(t^{n-1}, t^{n+1}; L^2(\Omega))}$$
$$+ \left(\|\nabla u(t^{n+1})\|^2 + \|\nabla I[u(t^{n+1})]\|\right) \|\nabla u_{tt}\|_{L^2(t^{n-1}, t^{n+1}; L^2(\Omega))}^2\Big)\|\nabla v_h\|.$$

Considering equation (33) and Lemma 4 , using equations (34)-(39), dividing both sides by $\|\nabla v_h\|$ and taking a supremum over V_h gives

(40)
$$\left\|\frac{D[\phi_h^{n+1}]}{\Delta t}\right\|_{X_h^*} \le C\Big[\Delta t^{-\frac{1}{2}} \|\eta_t\|_{L^2(t^n, t^{n+1}; L^2(\Omega))}\|$$
$$+ \|\nabla I[e_u^{n+1}]\|(\|\nabla I[u(t^{n+1})]\| + \|\nabla(I[u_h^{n+1}])\| + 1)$$
$$+ \|p(t^{n+1}) - \lambda_h^{n+1}\| + \Delta t^{\frac{3}{2}} \Big(\|u_{ttt}\|_{L^2(t^{n-1}, t^{n+1}; L^2(\Omega))} + \|\nabla u_{tt}\|_{L^2(t^{n-1}, t^{n+1}; L^2(\Omega))}$$
$$+ \|\nabla u\|_{L^4(t^{n-1}, t^{n+1}; L^2(\Omega))}^2 + \|\nabla u_{tt}\|_{L^4(t^{n-1}, t^{n+1}; L^2(\Omega))}^2\Big)\Big].$$

Reconsidering (33), we separate the pressure error term $e_p^{n+1} = (p(t^{n+1}) - \lambda_h^{n+1}) - (p_h^{n+1} - \lambda_h^{n+1})$ and rearrange, which yields for all $v_h \in X_h$

$$\left(p_h^{n+1} - \lambda_h^{n+1}, \nabla \cdot v_h\right) = -\left(\frac{D[\eta^{n+1}]}{\Delta t}, v_h\right) - \left(\frac{D[\phi^{n+1}]}{\Delta t}, v_h\right)$$

$$-b(I[e_u^{n+1}], I[u(t^{n+1})], v_h) - b(I[u_h^{n+1}], I[e_u^{n+1}], v_h) - \nu\left(\nabla I[e_u^{n+1}], \nabla v_h\right)$$

$$+ \left(p(t^{n+1}) - \lambda_h^{n+1}, \nabla \cdot v_h\right) + \tau^{n+1}(u, p; v_h).$$

Consider the estimates in (34)-(40). Divide by $\|\nabla v_h\|$, take the supremum over $v_h \in X_h$ and use the discrete inf-sup condition to obtain,

(41)

$$\beta\|p_h^{n+1} - \lambda_h^{n+1}\| \leq C\Big[\Delta t^{-\frac{1}{2}} \|\eta_t\|_{L^2(t^n, t^{n+1}; L^2(\Omega))}$$

$$+ \|\nabla I[e_u^{n+1}]\|\Big(\|\nabla I[u(t^{n+1})]\| + \|\nabla(I[u_h^{n+1}])\| + 1\Big)$$

$$+ \|p(t^{n+1}) - \lambda_h^{n+1}\| + \Delta t^{\frac{3}{2}}\Big(\|u_{ttt}\|_{L^2(t^{n-1}, t^{n+1}; L^2(\Omega))} + \|\nabla u_{tt}\|_{L^2(t^{n-1}, t^{n+1}; L^2(\Omega))}$$

$$+ \|\nabla u\|_{L^4(t^{n-1}, t^{n+1}; L^2(\Omega))}^2 + \|\nabla u_{tt}\|_{L^4(t^{n-1}, t^{n+1}; L^2(\Omega))}^2\Big)\Big].$$

We multiply by Δt, sum from $n = 1$ to $n = N - 1$ and apply triangle inequality. This yields

(42)

$$\beta\Delta t \sum_{n=1}^{N-1} \|e_p^{n+1}\| \leq C\Big[\Delta t^{-\frac{1}{2}} \|\eta_t\|_{L^2(0, T; L^2(\Omega))}$$

$$+ \||p - \lambda_h\||_{2,0} + \||\nabla I[e_u^{n+1}]\||_{2,0}$$

$$+ \Delta t^{\frac{5}{2}}\Big(\|u_{ttt}\|_{2,0}^2 + \|\nabla u_{tt}\|_{2,0} + \||\nabla u\||_{4,0}^2 + \|\nabla u_{tt}\|_{4,0}^2\Big)\Big].$$

Results from the equations $(A.21)$ and $(A.24)$ give the bounds for the first two terms. Using error estimates of the velocity on the third term and taking infimum over X_h and Q_h yield the result.   □

6. Numerical tests

We verify second order convergence for the new method through an exact solution in Section 6.1. Visualizations of the flow and benchmark quantities gives additional support to the increased accuracy of the new method in Section 6.3. The tests used P_2/P_1 and P_3/P_2 elements. All computations were performed with FEniCS [1].

6.1. Taylor-Green vortex . We apply the backward Euler and the backward Euler plus filter for the 2D Taylor-Green vortex. This test problem is historically used to assess accuracy and convergence rates in CFD [7]. The exact solution is given by

$$u = e^{-2\nu t}(\cos x \sin y, -\sin x \cos y) \text{ and } p = -\frac{1}{4}e^{-4\nu t}(\cos 2x + \cos 2y).$$

To test time accuracy, we solve using P_3/P_2 elements on a uniform mesh of 250×250 squares divided into 2 triangle per square. We take a series of time steps for which the total error is expected to be dominated by the temporal error. Since the true solution decays exponentially, we tabulate and display relative errors. Figure 2 displays the relative errors for backward Euler, backward Euler plus filtering only the velocity (Algorithm 1.1A), and backward Euler plus filtering both the velocity and pressure (Algorithm 1.1B). Filtering the pressure does not affect the velocity
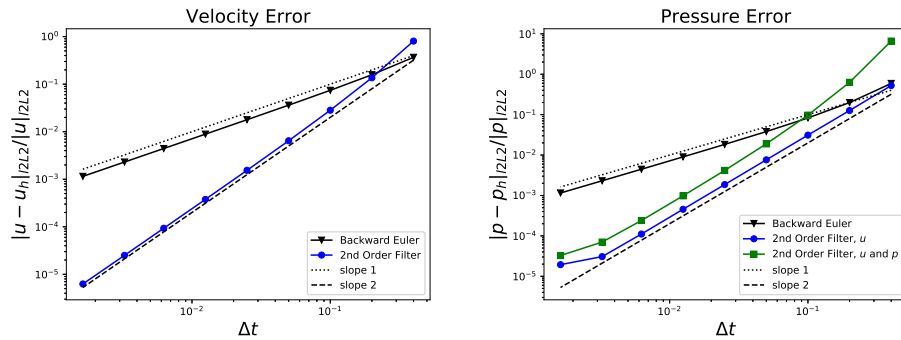
FIGURE 2. Convergence rates for the filtered quantities are second order as predicted. Filtering only the velocity produces the best pressure.

solution, so the velocity error plot only shows two lines. The velocity error is $\mathcal{O}(\Delta t^2)$, as predicted, and significantly smaller than the backward Euler error. Thus, adding the filter step (1.3) reduces the velocity error substantially, Figure 2, at negligible cost, Figure 1b. The pressure error is $\mathcal{O}(\Delta t^2)$ when either both u and p are filtered, or only u is filtered, which is consistent with our theoretical analysis. Filtering only u has smaller pressure error since the pressure filter introduces an extra consistency error term, see (17).

6.2. Adaptive Test. We test the time/order adaptive algorithm on a problem that showcases the superiority of the VSVO method over the constant stepsize, constant order method.

The Taylor-Green problem can be modified by replacing F with any differentiable function of t. With velocity and pressure defined as before, the required body force is

$$f(x, y, t) = (2\nu F(t) + F'(t))\langle \cos x \sin y, -\cos y \sin x \rangle.$$

For $F(t)$, we construct a sharp transition function between 0 and 1. First, let

$$g(t) = \begin{cases} 0 & \text{if } t \leq, 0 \\ \exp\left(-\frac{1}{(10t)^{10}}\right) & \text{if } t > 0. \end{cases}$$

This is a differentiable function, and $g(5) \equiv 1$ in double precision. Therefore, a differentiable (up to machine precision) function can be constructed with shifts and reflections of this function. This creates sections of flatness, and sections that rapidly change which require adaptivity to resolve efficiently. See Figure 3 for the evolution of $\|u\|$ with time. All tests were initialized at rest spaced at a constant interval of $\Delta t = 0.1$, 100 nodes per side of the square using P_2/P_1 elements, and with final time of 45.

Figure 3 compares two numerical solutions. One is from Algorithm 1.1 (second order - nonadaptive), and the other is from Algorithm 2.1 (VSVO-12). With $TOL = 10^{-3}$, the VSVO-12 method takes 342 steps, which comprises 254 accepted steps, and 88 rejected steps. The constant stepsize method which took 535 steps does not accurately capture the energetic jumps.

Figure 4 shows the relative l^2L^2 velocity errors versus steps taken of VSVO-12 for seven different TOLs, starting at 10^{-1}, and dividing by ten down to 10^{-7}. This is compared with nonadaptive method (which has no rejected steps) sampled at several stepsizes. Both methods show second order convergence, but for smaller tolerances, VSVO-12 performs about 10^3 better than the nonadaptive method for the same amount of work.
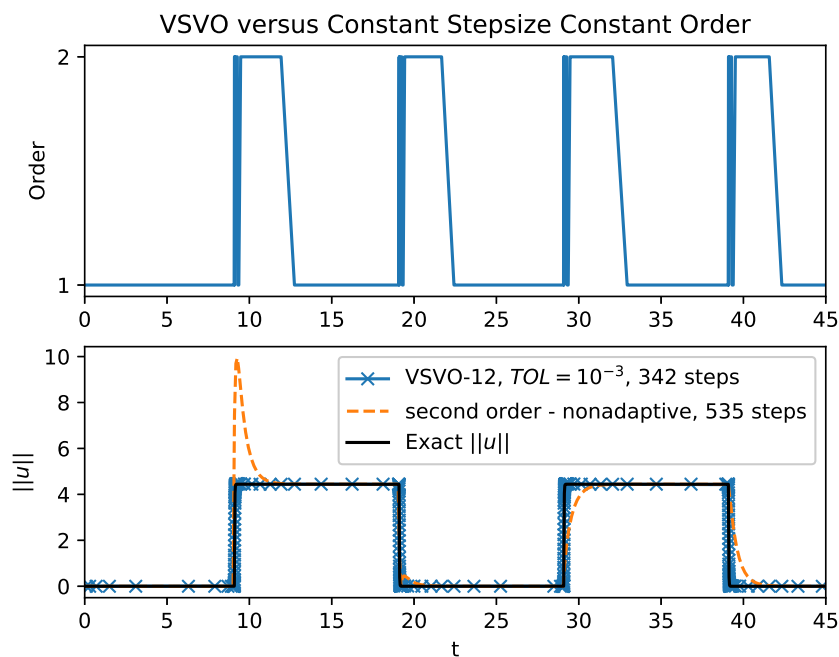

FIGURE 3. The nonadaptive second order method results in large overshoots and undershoots while requiring more work than the adaptive method.

6.3. Flow around a cylinder. We now use the benchmark problem of flow around a cylinder, originally proposed in [31], to test the improvement obtained using filters on flow quantities (drag, lift, and pressure drop) using values obtained via a DNS in [24] as a reference. This problem has also been used as a benchmark in [28],[27],[5],[6] and others. Let $\nu = 10^{-3}$, $f \equiv 0$, $T_{final} = 8$, and

$$\Omega = \{(x,y) \,|\, 0 < x < 2.2, \;\; 0 < y < 0.41 \text{ and } (x-0.2)^2 + (y-0.2)^2 > 0.05^2\},$$

i.e., a channel with a cylindrical cutout. A parabolic velocity of $u = 0.41^{-2}\sin(\pi t/8)$ $\times(6y(0.41 - y), 0)$ is prescribed at the left and right boundaries. We used a spatial discretization with 479026 degrees of freedom with 1000 vertices on the boundary of the cylinder. The mesh used $P2/P1$ elements, and was obtained by adaptive refinement from solving the steady solution with $u = 0.41^{-2}(6y(0.41 - y), 0)$ as inflow and outflow boundary conditions.
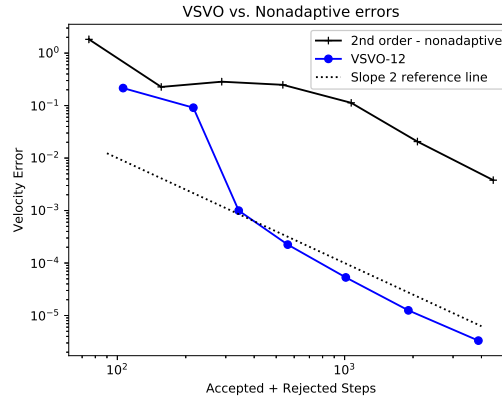
FIGURE 4. The VSVO-12 method performs three orders of magnitude better for the same amount of work compared to the nonadaptive 2nd order method for the test problem in Section 6.2. Each circle represents a different tolerance from $TOL = 10^{-1}$ to 10^{-7}.
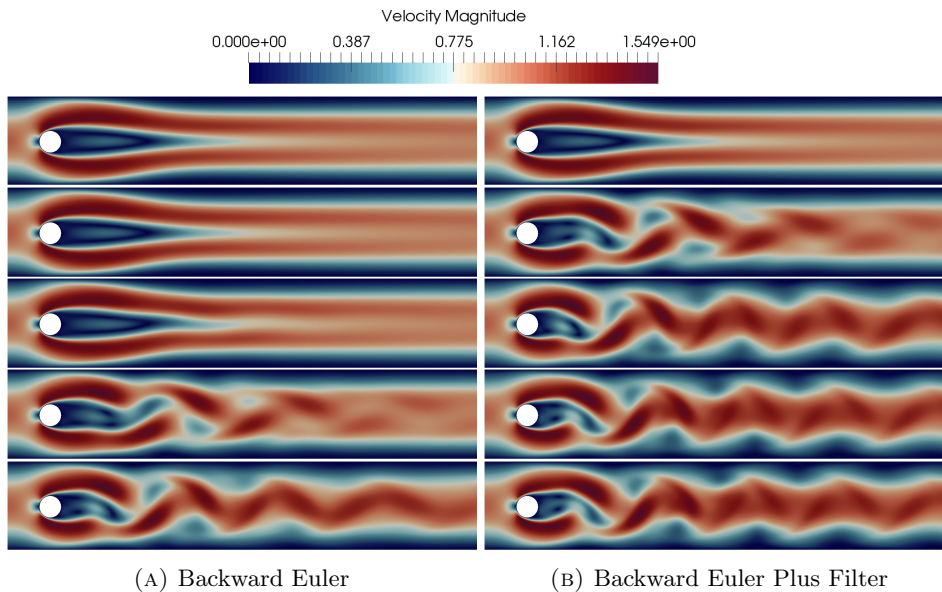

(A) Backward Euler   (B) Backward Euler Plus Filter

FIGURE 5. Flow snapshots at $t = 6$ with $\Delta t = 0.04$ (top), and Δt halving until $\Delta t = 0.0025$ (bottom). Backward Euler (left) destroys energy and suppresses oscillations, meaning that it can predict nearly steady state solutions when a time dependent one exists. The time filter (right) corrects this.

The correct behavior for this problem is that vortices shed off the cylinder as the inlet and outlet velocities increase. Figure 5 shows snapshots of the flow at $t = 6$ for

five successively halved Δt's. The Backward Euler approximation shows no vortex shedding for $\Delta t = 0.04, 0.02$, and 0.01. The filtered method of Algorithm 1.1 shows the qualitatively correct behavior from $\Delta t = 0.02$ on. Clearly, higher order and less dissipative methods are necessary to see dynamics for modestly large Δt.

It was demonstrated in [24] that the backward Euler time discretization greatly under predicts lift except for very small step sizes. Figure 6 demonstrates that the time filter in Algorithm 1.1 corrects both the amplitude and phase error in the backward Euler approximation. Other quantities that were compared to reference values were the maximum drag $c_{d,\max}$, the time of max drag $t(c_{d,\max})$, time of maximum lift $t(c_{l,\max})$, and pressure drop across the cylinder at $t = 8$ are shown in Table 1.

The choice of whether or not to filter the pressure does not affect the velocity solution, the snapshots shown Figure 5 are the same for both choices. Table 1 shows that filtering u greatly improves the calculated flow quantities whether or not p is filtered.
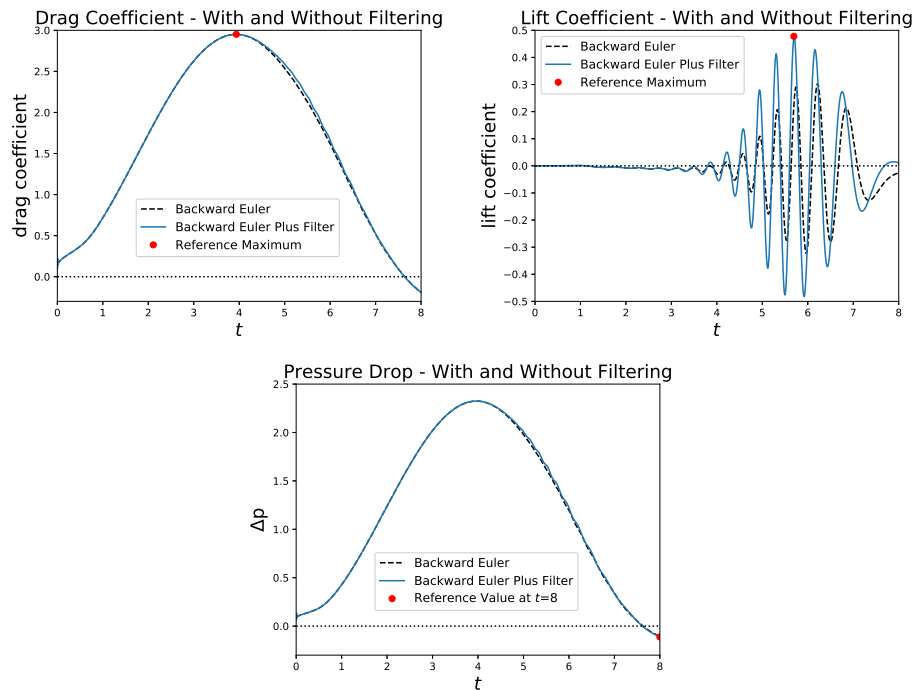

FIGURE 6. Lift of the Backward Euler solution and the filtered solution for $\Delta t = 0.0025$. The filtered solution correctly predicts both the time and magnitude of the maximum lift.

7. Conclusion

Accurate and stable time discretization is important for obtaining correct flow predictions. The backward Euler time discretization is a stable but inaccurate method. We have shown that for minimum extra programming effort, computational complexity, and storage, second order accuracy and unconditional stability can

TABLE 1. Lift, drag, and pressure drop for cylinder problem.

Δt	$t(c_{d,\max})$	$c_{d,\max}$	$t(c_{l,\max})$	$c_{l,\max}$	$\Delta p(8)$
Backward Euler					
0.04	3.92	2.95112558	0.88	0.00113655	-0.12675521
0.02	3.94	2.95064522	0.92	0.00117592	-0.12647232
0.01	3.93	2.95041574	7.17	0.02489640	-0.12433915
0.005	3.93	2.95031983	6.28	0.17588270	-0.10051423
0.0025	3.9325	2.95038901	6.215	0.30323034	-0.10699361
Backward Euler Plus Filter					
0.04	3.92	2.95021463	7.56	0.00438111	-0.12628328
0.02	3.94	2.95026781	6.14	0.20559211	-0.11146505
0.01	3.93	2.95060684	5.81	0.40244197	-0.09943203
0.005	3.935	2.95082513	5.72	0.46074771	-0.11111586
0.0025	3.935	2.95089028	5.7	0.47414096	-0.11193754
Backward Euler Plus Filter u and p					
0.04	3.92	2.95073993	7.52	0.00439864	-0.12642684
0.02	3.94	2.95039973	6.14	0.21101313	-0.11153593
0.01	3.93	2.95063962	5.81	0.40624697	-0.09945143
0.005	3.935	2.95083296	5.72	0.46192306	-0.11112049
0.0025	3.935	2.95089220	5.7	0.47444753	-0.11193859
Reference Values					
—	3.93625	2.950921575	5.693125	0.47795	-0.1116

be obtained by adding a time filter. Due to the embedded and modular structure of the algorithm, both adaptive time-step and adaptive order are easily implemented in a code based on a backward Euler time discretization. Extension of the method and analysis to yet higher order time discretization is important as is exploring the effect of time filters on other methods possible for Step 1 of Algorithm 1.1. Analysis of the effect of time filters with moving and time dependent boundary conditions would also be a significant extension.

Acknowledgements

References

[1] M. Alnæs, J. Blechta, J. Hake, A. Johansson, B. Kehlet, A. Logg, C. Richardson, J. Ring, M. Rognes, and G. Wells. The FEniCS project version 1.5. *Archive of Numerical Software*, 3(100), 2015.

[2] D. N. Arnold, F. Brezzi, and M. Fortin. A stable finite element for the Stokes equations. *CALCOLO*, 21(4):337–344, Dec 1984.

[3] R. Asselin. Frequency filter for time integrations. *Mon. Weather Rev*, (100):487–490, 1972.

[4] G. A. Baker, V. A. Dougalis, and O. A. Karakashian. On a higher order accurate fully discrete Galerkin approximation to the Navier-Stokes equations. *Mathematics of Computation*, 39(160):339–375, 1982.

[5] M. Besier and R. Rannacher. Goal-oriented spacetime adaptivity in the finite element Galerkin method for the computation of nonstationary incompressible flow. *International Journal for Numerical Methods in Fluids*, 70(9):1139–1166, January 2012.

[6] S. Charnyi, T. Heister, M. A. Olshanskii, and L. G. Rebholz. On conservation laws of Navier-Stokes Galerkin discretizations. *Journal of Computational Physics*, 337:289 – 308, 2017.

[7] A. J. Chorin. The numerical solution of the Navier-Stokes equations for an incompressible fluid. *Bull. Amer. Math. Soc.*, 73(6):928–931, 11 1967.

[8] M. Crouzeix and P. A. Raviart. Approximation d'équations d'évolution linéaires par des méthodes multipas. *Etude Numérique des Grands Systèmes*, 1976.

[9] G. Dahlquist. G-stability is equivalent to A-stability. *BIT Numerical Mathematics*, 18(4):384–401, Dec 1978.

[10] E. Emmrich. Error of the two-step BDF for the incompressible Navier-Stokes problem. *E-SAIM: M2AN*, 38(5):757–764, 2004.

[11] E. Emmrich. Stability and convergence of the two-step BDF for the incompressible Navier-Stokes problem. 5:199–209, 01 2004.

[12] J. A. Fiordilino. On pressure estimates for the Navier-Stokes equations. *ArXiv e-prints*, Mar. 2018.

[13] K. J. Galvin. New subgrid artificial viscosity Galerkin methods for the Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 200(1):242 – 250, 2011.

[14] T. Geveci. On the convergence of a time discretization scheme for the Navier-Stokes equations. *Mathematics of Computation*, 53(187):43–53, 1989.

[15] V. Girault and P.-A. Raviart. *Finite Element Approximation of the Navier-Stokes Equations*. Springer-Verlag Berlin Heidelberg, 1979.

[16] V. Girault and P.-A. Raviart. An optimally accurate discrete regularization for second order timestepping methods for navier–stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 310:388–405, 2016.

[17] P. M. Gresho and R. L. Sani. *Incompressible Flow and the Finite Element Method*. John Wiley & Sons, Inc., 1998.

[18] D. Griffiths and D. Higham. Numerical methods for ordinary differential equations. *Springer*, 2010.

[19] D. F. Griffiths and D. J. Higham. *Numerical Methods for Ordinary Differential Equations*. Springer-Verlag London Limited, 2010.

[20] A. Guzel and W. Layton. Time filters increase accuracy of the fully implicit method. *BIT Numerical Mathematics*, 58(2):301–315, Jun 2018.

[21] J. Heywood and R. Rannacher. Finite element approximation of the nonstationary Navier-Stokes problem. Part IV. Error analysis for second-order time discretization. *SIAM Journal on Numerical Analysis*, 27(2):353384, 1990.

[22] N. Jiang. A second-order ensemble method based on a blended backward differentiation formula timestepping scheme for time-dependent Navier-Stokes equations. *Numerical Methods for Partial Differential Equations*, 33(1):34–61.

[23] N. Jiang, M. Mohebujjaman, L. G. Rebholz, and C. Trenchea. An optimally accurate discrete regularization for second order timestepping methods for Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 310:388 – 405, 2016.

[24] V. John. Reference values for drag and lift of a two-dimensional time-dependent flow around a cylinder. *International Journal for Numerical Methods in Fluids*, 44(7):777–788, 2004.

[25] W. Layton, N. Mays, M. Neda, and C. Trenchea. Numerical analysis of modular regularization methods for the BDF2 time discretization of the Navier-Stokes equations. *ESAIM: M2AN*, 48(3):765–793, 2014.

[26] W. Layton and C. Trenchea. Stability of two IMEX methods, CNLF and BDF2-AB2, for uncoupling systems of evolution equations. *Applied Numerical Mathematics*, 62:112–120, 2012.

[27] J.-G. Liu, J. Liu, and R. L. Pego. Stable and accurate pressure approximation for unsteady incompressible viscous flow. *Journal of Computational Physics*, 229(9):3428 – 3453, 2010.

[28] J. Rang and L. Angermann. New Rosenbrock W-Methods of order 3 for partial differential algebraic equations of index 1. *BIT Numerical Mathematics*, 45(4):761–787, Dec 2005.

[29] S. S. Ravindran. An extrapolated second order backward difference time-stepping scheme for the magnetohydrodynamics system. *Numerical Functional Analysis and Optimization*, 37(8):990–1020, 2016.

[30] A. J. Robert. *An evaluation of the behaviour of planetary waves in an atmospheric model based on spherical harmonics.* PhD thesis, McGill, 1965.

[31] M. Schäfer and S. Turek. Benchmark computations of laminar flow around a cylinder. In H. E.H., editor, *Flow Simulation with High-Performance Computers II. Notes on Numerical Fluid Mechanics*, volume 48, pages 547–566. Vieweg+Teubner Verlag, 1996.

[32] R. Temam. *Navier-Stokes Equations and Nonlinear Functional Analysis.* Society for Industrial and Applied Mathematics, 1995.

[33] R. Verfürth. Error estimates for a mixed finite element approximation of the Stokes equations. *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique*, 18(2):175–182, 1984.

[34] P. D. Williams. A proposed modification to the Robert-Asselin time filter. *Monthly Weather Review*, 137:2538–2546, 08 2009.

Appendix A. Velocity Error Analysis

A.1. Proof of Lemma 9.

Proof. By Taylor's theorem with the integral remainder,

$$D[u(t^{n+1})] - \Delta t u_t(t^{n+1}) = \frac{3}{2}u(t^{n+1}) - \Delta t u_t(t^{n+1})$$

$$-2\left(u(t^{n+1}) - \Delta t u_t(t^{n+1}) + \frac{\Delta t^2}{2}u_{tt}(t^{n+1})) + \frac{1}{2}\int_{t^{n+1}}^{t^n} u_{ttt}(t)(t^n - t)^2 dt\right)$$

$$+\frac{1}{2}\left(u(t^{n+1}) - 2\Delta t u_t(t^{n+1}) + 2\Delta t^2 u_{tt}(t^{n+1})) + \frac{1}{2}\int_{t^{n+1}}^{t^{n-1}} u_{ttt}(t)(t^{n-1} - t)^2 dt\right)$$

$$= \int_{t^n}^{t^{n+1}} u_{ttt}(t^n - t)^2 dt - \frac{1}{4}\int_{t^{n-1}}^{t^{n+1}} u_{ttt}(t^{n-1} - t)^2 dt.$$

These terms are first estimated by Cauchy-Schwarz.

$$\left(\int_{t^n}^{t^{n+1}} u_{ttt}(t)(t^n - t)^2 dt\right)^2 \leq \int_{t^n}^{t^{n+1}} u_{ttt}^2 dt \int_{t^n}^{t^{n+1}} (t^n - t)^4 dt = \frac{\Delta t^5}{5}\int_{t^n}^{t^{n+1}} u_{ttt}^2 dt.$$

$$\frac{1}{16}\left(\int_{t^{n-1}}^{t^{n+1}} u_{ttt}(t)(t^{n-1} - t)^2 dt\right)^2 \leq \frac{1}{16}\int_{t^{n-1}}^{t^{n+1}} u_{ttt}^2 dt \int_{t^{n-1}}^{t^{n+1}} (t^{n-1} - t)^4 dt$$

$$(A.1) \qquad\qquad = \frac{2\Delta t^5}{5}\int_{t^{n-1}}^{t^{n+1}} u_{ttt}^2 dt.$$

Thus,

$$\left(\frac{D[u(t^{n+1})]}{\Delta t} - u_t(t^{n+1})\right)^2 \leq \frac{6}{5}\Delta t^3 \int_{t^{n-1}}^{t^{n+1}} u_{ttt}^2 dt.$$

Integrating with respect to x yields the first inequality. Next,

$$I[u(t^{n+1})] - u(t^{n+1}) = \frac{1}{2}u(t^{n+1}) - u(t^n) + \frac{1}{2}u(t^{n-1})$$

$$= \frac{1}{2}\int_{t^n}^{t^{n+1}} u_{tt}(t)(t^{n+1} - t)dt + \frac{1}{2}\int_{t^n}^{t^{n-1}} u_{tt}(t)(t^{n-1} - t)dt.$$

By similar steps,

$$\left(\int_{t^n}^{t^{n+1}} u_{tt}(t)(t^{n+1} - t)dt \right)^2 \leq \frac{\Delta t^3}{3} \int_{t^n}^{t^{n+1}} u_{tt}^2 dt.$$

$$\left(\int_{t^{n-1}}^{t^n} u_{tt}(t)(t^{n-1} - t)dt \right)^2 \leq \frac{\Delta t^3}{3} \int_{t^{n-1}}^{t^n} u_{tt}^2 dt.$$

Therefore,

$$(A.2) \qquad \left(I[u(t^{n+1})] - u(t^{n+1}) \right)^2 \leq \frac{\Delta t^3}{6} \int_{t^{n-1}}^{t^{n+1}} u_{tt}^2 dt.$$

The last inequality can be proved using the same strategy. $\qquad \square$

A.2. Proof of Theorem 10.

Proof. We prove this for Option A. A parallel proof exists for Option B. At $t^{n+1} = (n+1)\Delta t$, the true solution of (1) satisfies,

$$(A.3) \quad \begin{aligned} &\left(\frac{D[u(t^{n+1})]}{\Delta t}, v_h \right) + b\left(I[u(t^{n+1})], I[u(t^{n+1})], v_h \right) \\ &+ \nu \left(\nabla I[u(t^{n+1})], \nabla v_h \right) - \left(p(t^{n+1}), \nabla \cdot v_h \right) \\ &= \left(\mathbf{f}^{n+1}, v_h \right) + \tau^{n+1}(u, p; v_h) \quad \forall v_h \in X_h. \end{aligned}$$

Subtracting (12) from (A.3) yields

$$(A.4) \quad \begin{aligned} &\left(\frac{D[e_u^{n+1}]}{\Delta t}, v_h \right) + b\left(I[e_u^{n+1}], I[u(t^{n+1})], v_h \right) \\ &+ b\left(I[u_h^{n+1}], I[e_u^{n+1}], v_h \right) + \nu \left(\nabla I[e_u^{n+1}], \nabla v_h \right) \\ &- \left(e_p^{n+1}, \nabla \cdot v_h \right) = \tau^{n+1}(u, p; v_h). \end{aligned}$$

Decompose the error equation for velocity

$$(A.5) \qquad u(t_{n+1}) - u_h^{n+1} = (u^{n+1} - \tilde{u}_h^{n+1}) + (\tilde{u}_h^{n+1} - u_h^{n+1}) = \eta^{n+1} + \phi_h^{n+1}.$$

where $\tilde{u}_h^{n+1} \in V_h$ is the best approximation of $u(t^{n+1})$ in V_h. Similarly, we let $\lambda_h^{n+1} \in Q_h$ be the best approximation of $p(t^{n+1})$ in Q_h. That is, they minimize $\|u(t^{n+1}) - v\|_X$ and $\|p(t^{n+1}) - q\|$, respectively.

Set $v_h = I[\phi_h^{n+1}]$. Using the identity (10) with $a = \phi_h^{n+1}$, $b = \phi_h^n$, $c = \phi_h^{n-1}$, (A.5), and applying $(\lambda_h, \nabla \cdot \phi_h) = 0$ for all $\lambda_h \in Q_h$, equation (A.4) can be written

$$(A.6) \quad \begin{aligned} &\frac{1}{4\Delta t}(\|\phi_h^{n+1}\|^2 + \|2\phi_h^{n+1} - \phi_h^n\|^2 + \|\phi_h^{n+1} - \phi_h^n\|^2) \\ &- \frac{1}{4\Delta t}(\|\phi_h^n\|^2 + \|2\phi_h^n - \phi_h^{n-1}\|^2 + \|\phi_h^n - \phi_h^{n-1}\|^2) \\ &+ \frac{3}{4\Delta t}\|\phi_h^{n+1} - 2\phi_h^n + \phi_h^{n-1}\|^2 + \nu\|\nabla I[\phi_h^{n+1}]\|^2 \\ &= -\left(\frac{D[\eta^{n+1}]}{\Delta t}, I[\phi_h^{n+1}] \right) - b\left(I[\phi_h^{n+1}], I[u(t^{n+1})], I[\phi_h^{n+1}] \right) \\ &- b\left(I[u_h^{n+1}], I[\eta^{n+1}], I[\phi_h^{n+1}] \right) - b\left(I[\eta^{n+1}], I[u(t^{n+1})], I[\phi_h^{n+1}] \right) \\ &+ \left(p(t^{n+1}) - \lambda_h^{n+1}, \nabla \cdot I[\phi_h^{n+1}] \right) - \nu \left(\nabla I[\eta^{n+1}], \nabla I[\phi_h^{n+1}] \right) \\ &+ \tau^{n+1}(u, p; I[\phi_h^{n+1}]). \end{aligned}$$

The next step in the proof is to bound all the terms on the right hand side of $(A.6)$ and absorb terms into the left hand side. For arbitrary $\varepsilon > 0$, the first term on the right hand side of (A.6) is bounded in the following way,

$$(A.7) \qquad -\left(\frac{D[\eta^{n+1}]}{\Delta t}, I[\phi_h^{n+1}]\right) \le \frac{1}{4\varepsilon}\left\|\frac{D[\eta^{n+1}]}{\Delta t}\right\|_{-1}^2 + \varepsilon\|\nabla I[\phi_h^{n+1}]\|^2.$$

The first nonlinear term can be bounded as

$$(A.8) \qquad \begin{aligned} &-b\left(I[\phi_h^{n+1}], I[u(t^{n+1})], I[\phi_h^{n+1}]\right) \le C\|I[\phi_h^{n+1}]\|\|I[u(t^{n+1})]\|_2\|\nabla I[\phi_h^{n+1}]\| \\ &\le \frac{C^2}{4\varepsilon}\|I[\phi_h^{n+1}]\|^2\|I[u(t^{n+1})]\|_2^2 + \varepsilon\|\nabla I[\phi_h^{n+1}]\|^2. \end{aligned}$$

The second nonlinear term is estimated by rewriting it using (A.5) as follows

$$(A.9) \qquad \begin{aligned} -b\left(I[u_h^{n+1}], I[\eta^{n+1}], I[\phi_h^{n+1}]\right) &= -b\left(I[u(t^{n+1})], I[\eta^{n+1}], I[\phi_h^{n+1}]\right) \\ &+ b\left(I[\eta^{n+1}], I[\eta^{n+1}], I[\phi_h^{n+1}]\right) + b\left(I[\phi_h^{n+1}], I[\eta^{n+1}], I[\phi_h^{n+1}]\right). \end{aligned}$$

then find bounds for all terms on the right hand side of (A.9). We bound the third nonlinear term in (A.6) the same way as the first nonlinear term in (A.9).

$$(A.10) \qquad \begin{aligned} &-b\left(I[u(t^{n+1})], I[\eta^{n+1}], I[\phi_h^{n+1}]\right) \\ &\le C\|\nabla I[u(t^{n+1})]\|\|\nabla I[\eta^{n+1}]\|\|\nabla I[\phi_h^{n+1}]\| \\ &\le \frac{C^2}{4\varepsilon}\|u\|_{\infty,1}^2\|\nabla I[\eta^{n+1}]\|^2 + \varepsilon\|\nabla I[\phi_h^{n+1}]\|^2, \end{aligned}$$

and

$$(A.11) \qquad b\left(I[\eta^{n+1}], I[\eta^{n+1}], I[\phi_h^{n+1}]\right) \le \frac{C^2}{4\varepsilon}\|\nabla I[\eta^{n+1}]\|^4 + \varepsilon\|\nabla I[\phi_h^{n+1}]\|^2.$$

Next, by the inverse inequality, approximation assumptions, and Lemma 3,

$$(A.12) \qquad \begin{aligned} &b\left(I[\phi_h^{n+1}], I[\eta^{n+1}], I[\phi_h^{n+1}]\right) \\ &\le C\|\nabla I[\phi_h^{n+1}]\|\|\nabla I[\eta^{n+1}]\|\|\nabla I[\phi_h^{n+1}]\| \\ &\le Ch^{-1}\|I[\phi_h^{n+1}]\|\|\nabla I[\eta^{n+1}]\|\|\nabla I[\phi_h^{n+1}]\| \\ &\le C\|I[\phi_h^{n+1}]\|\|I[u(t^{n+1})]\|_2\|\nabla I[\phi_h^{n+1}]\| \\ &\le \frac{C^2}{4\varepsilon}\|I[\phi_h^{n+1}]\|^2\|I[u(t^{n+1})]\|_2^2 + \varepsilon\|\nabla I[\phi_h^{n+1}]\|^2. \end{aligned}$$

The pressure can be bounded as follows
$$(A.13)$$
$$\left(p(t^{n+1}) - \lambda_h^{n+1}, \nabla \cdot I[\phi_h^{n+1}]\right) \le \frac{C^2}{4\varepsilon}\|p(t^{n+1}) - \lambda_h^{n+1}\|^2 + \varepsilon\|\nabla I[\phi_h^{n+1}]\|^2.$$

Then we can bound the term after the pressure,

$$(A.14) \qquad -\nu\left(\nabla I[\eta^{n+1}], \nabla(I[\phi_h^{n+1}])\right) \le \frac{\nu}{2}\|\nabla I[\eta^{n+1}]\|^2 + \frac{\nu}{2}\|\nabla I[\phi_h^{n+1}]\|^2.$$

Next we will bound all components of the consistency error $\tau^{n+1}(u, p; I[\phi_h^{n+1}])$.

(A.15)
$$\left(\frac{D[u(t^{n+1})]}{\Delta t} - u_t(t^{n+1}), I[\phi_h^{n+1}] \right)$$
$$\leq C \| \frac{D[u(t^{n+1})]}{\Delta t} - u_t(t^{n+1}) \| \| \nabla I[\phi_h^{n+1}] \|$$
$$\leq \frac{C^2}{4\varepsilon} \| \frac{D[u(t^{n+1})]}{\Delta t} - u_t(t^{n+1}) \|^2 + \varepsilon \| \nabla I[\phi_h^{n+1}] \|^2.$$

(A.16)
$$\nu \left(\nabla (I[u(t^{n+1})] - u(t^{n+1})), \nabla I[\phi_h^{n+1}] \right)$$
$$\leq \frac{C^2}{4\varepsilon} \| \nabla (I[u(t^{n+1})] - u(t^{n+1})) \|^2 + \varepsilon \| \nabla I[\phi_h^{n+1}] \|^2.$$

The nonlinear term in $\tau^{n+1}(u, p; I[\phi_h^{n+1}])$ is then estimated as follows,

(A.17)
$$b\left(I[u(t^{n+1})], I[u(t^{n+1})], I[\phi_h^{n+1}] \right) - b(u(t^{n+1}), u(t^{n+1}), I[\phi_h^{n+1}])$$
$$= b\left(I[u(t^{n+1})] - u(t^{n+1}), I[u(t^{n+1})], I[\phi_h^{n+1}] \right)$$
$$\quad + b(u(t^{n+1}), I[u(t^{n+1})] - u(t^{n+1}), I[\phi_h^{n+1}])$$
$$\leq C \| \nabla (I[u(t^{n+1})] - u(t^{n+1})) \| \| \nabla I[\phi_h^{n+1}] \| \left(\| \nabla I[u(t^{n+1})] \| + \| \nabla u(t^{n+1}) \| \right)$$
$$\leq \frac{C^2}{4\varepsilon} \| \nabla (I[u(t^{n+1})] - u(t^{n+1})) \|^2 \left(\| \nabla I[u(t^{n+1})] \|^2 + \| \nabla u(t^{n+1}) \|^2 \right)$$
$$\quad + \varepsilon \| \nabla I[\phi_h^{n+1}] \|^2.$$

Set $\varepsilon = \frac{\nu}{40}$. Using $(A.7)$ to $(A.14)$ in $(A.6)$ yields
(A.18)
$$\frac{1}{4\Delta t}(\| \phi_h^{n+1} \|^2 + \| 2\phi_h^{n+1} - \phi_h^n \|^2 + \| \phi_h^{n+1} - \phi_h^n \|^2) + \frac{\nu}{4} \| \nabla I[\phi_h^{n+1}] \|^2$$
$$- \frac{1}{4\Delta t}(\| \phi_h^n \|^2 + \| 2\phi_h^n - \phi_h^{n-1} \|^2 + \| \phi_h^n - \phi_h^{n-1} \|^2) + \frac{3}{4\Delta t} \| \phi_h^{n+1} - 2\phi_h^n + \phi_h^{n-1} \|^2$$
$$\leq C\nu^{-1}\Big(\| \frac{D[\eta^{n+1}]}{\Delta t} \|_{-1}^2 + \| I[\phi_h^{n+1}] \|^2 \| I[u(t^{n+1})] \|_2^2$$
$$+ \|u\|_{\infty,1} \| \nabla I[\eta^{n+1}] \|^2 + \| \nabla I[\eta^{n+1}] \|^4 + \| p(t^{n+1}) - \lambda_h^{n+1} \|^2$$
$$+ \nu^2 \| \nabla I[\eta^{n+1}] \|^2 + \| \frac{D[u(t^{n+1})]}{\Delta t} - u_t(t^{n+1}) \|^2$$
$$+ \| \nabla (I[u(t^{n+1})] - u(t^{n+1})) \|^2$$
$$+ \| \nabla (I[u(t^{n+1})] - u(t^{n+1})) \|^2 (\| \nabla I[u(t^{n+1})] \|^2 + \| \nabla u(t^{n+1}) \|^2) \Big).$$

Let $\kappa = C\frac{\|u\|_{\infty,2}^2}{\nu}$. Assume $\Delta t < \frac{1}{\kappa}$, summing from $n = 1$ to $n = N-1$ and applying the discrete Gronwall lemma we obtain

(A.19)
$$\|\phi_h^N\|^2 + \|2\phi_h^N - \phi_h^{N-1}\|^2 + \|\phi_h^N - \phi_h^{N-1}\|^2$$
$$+ \sum_{n=1}^{N-1} 3\|\phi_h^{n+1} - 2\phi_h^n + \phi_h^{n-1}\|^2 + \nu\Delta t \sum_{n=1}^{N-1} \|\nabla I[\phi_h^{n+1}]\|^2$$
$$\leq e^{\left(\frac{\Delta t \kappa (N-1)}{1 - \Delta t \kappa}\right)} \Bigg(\|\phi_h^1\|^2 + \|2\phi_h^1 - \phi_h^0\|^2 + \|\phi_h^1 - \phi_h^0\|^2 + C\Delta t \sum_{n=1}^{N-1} \|\frac{D[\eta^{n+1}]}{\Delta t}\|_{-1}^2$$
$$+ C\Delta t(\|u\|_{\infty,1}^2 + \nu^2) \sum_{n=1}^{N-1} \|\nabla I[\eta^{n+1}]\|^2 + C\Delta t \sum_{n=1}^{N-1} \|\nabla I[\eta^{n+1}]\|^4$$
$$+ C\Delta t \sum_{n=1}^{N-1} \|p(t^{n+1}) - \lambda_h^{n+1}\|^2 + C\Delta t \sum_{n=1}^{N-1} \|\frac{D[u(t^{n+1})]}{\Delta t} - u_t(t^{n+1})\|^2$$
$$+ C\Delta t \sum_{n=1}^{N-1} \|\nabla(I[u(t^{n+1})] - u(t^{n+1}))\|^2$$
$$+ C\Delta t \sum_{n=1}^{N-1} \|\nabla(I[u(t^{n+1})] - u(t^{n+1}))\|^2 (\|\nabla I[u(t^{n+1})]\|^2 + \|\nabla u(t^{n+1})\|^2) \Bigg).$$

The first three terms can be bounded as

(A.20)
$$\|\phi_h^1\|^2 + \|2\phi_h^1 - \phi_h^0\|^2 + \|\phi_h^1 - \phi_h^0\|^2$$
$$\leq C\Big(\|u(t_1) - u_h^1\|^2 + \|(u(t_0) - u_h^0)\|^2 \Big) + Ch^{2k+2}\|u\|_{\infty,k+1}.$$

We bound the fourth term in (A.19) as follows

(A.21)
$$\Delta t \sum_{n=1}^{N-1} \|\frac{D[\eta^{n+1}]}{\Delta t}\|_{-1}^2 = \Delta t \sum_{n=1}^{N-1} \|\frac{\frac{3}{2}(\eta^{n+1} - \eta^n) - \frac{1}{2}(\eta^n - \eta^{n-1})}{\Delta t}\|_{-1}^2$$
$$\leq C \sum_{n=0}^{N} \int_{t^{n-1}}^{t^{n+1}} \|\eta_t\|^2 ds \leq Ch^{2k}\|u_t\|_{2,k+1}^2,$$

and

(A.22)
$$\Delta t(\|u\|_{\infty,1}^2 + \nu^2) \sum_{n=1}^{N-1} \|\nabla I[\eta^{n+1}]\|^2$$
$$\leq C\Delta t(\|u\|_{\infty,1}^2 + \nu^2) \max\left\{\frac{9}{4}, 1, \frac{1}{4}\right\} \sum_{n=1}^{N-1} 3\left(\|\nabla \eta^{n+1}\|^2 + \|\nabla \eta^n\|^2 + \|\nabla \eta^{n-1}\|^2\right)$$
$$\leq C\Delta t \sum_{n=0}^{N} h^{2k}\|u^{n+1}\|_{k+1}^2 = Ch^{2k}\|u\|_{2,k+1}^2.$$

Similarly to (A.22), we also have

(A.23)
$$\Delta t \sum_{n=1}^{N-1} \|\nabla I[\eta^{n+1}]\|^4 \leq C\Delta t \sum_{n=0}^{N} h^{4k}\|u(t^{n+1})\|_{k+1}^4 = Ch^{4k}\|u\|_{4,k+1}^4.$$

Observe that

$$(A.24) \qquad \Delta t \sum_{n=1}^{N} \|p(t^{n+1}) - \lambda_h^{n+1}\|^2 \leq Ch^{2s+2} \|\|p\|\|_{2,s+1}^2.$$

The terms from consistency error are bounded using Lemma 9.

$$\Delta t \sum_{n=1}^{N-1} \|\frac{D[u(t^{n+1})]}{\Delta t} - u_t(t^{n+1})\|^2 \leq C\Delta t^4 \sum_{n=0}^{N-1} \int_{t^{n-1}}^{t^{n+1}} \|u_{ttt}\|^2 dt$$
$$(A.25) \qquad\qquad\qquad\qquad\qquad\qquad \leq C\Delta t^4 \|u_{ttt}\|_{2,0}^2.$$

$$\Delta t \sum_{n=1}^{N-1} \|\nabla(I[u(t^{n+1})] - u(t^{n+1}))\|^2 \leq C\Delta t^4 \sum_{n=1}^{N-1} \int_{t^{n-1}}^{t^{n+1}} \|\nabla u_{tt}\|^2 dt$$
$$(A.26) \qquad\qquad\qquad\qquad\qquad\qquad \leq C\Delta t^4 \|\nabla u_{tt}\|_{2,0}^2.$$

$$\Delta t \sum_{n=1}^{N-1} \|\nabla(I[u(t^{n+1})] - u(t^{n+1}))\|^2 (\|\nabla I[u(t^{n+1})]\|^2 + \|\nabla u(t^{n+1})\|^2)$$
$$(A.27) \qquad \leq C\Delta t \sum_{n=1}^{N-1} (\|\nabla I[u(t^{n+1})]\|^2 + \|\nabla u(t^{n+1})\|^2)\Delta t^3 \int_{t^{n-1}}^{t^{n+1}} \|\nabla u_{tt}\|^2 dt$$
$$\leq C\Delta t^4 \|\nabla u\|_{\infty,0} \sum_{n=1}^{N-1} \|\nabla u_{tt}\|^2 dt = C\Delta t^4 \|u_{tt}\|_{2,1}^2.$$

Combining $(A.20)$ - $(A.27)$ gives

$$\|\phi_h^N\|^2 + \|2\phi_h^N - \phi_h^{N-1}\|^2 + \|\phi_h^N - \phi_h^{N-1}\|^2$$
$$+ \sum_{n=1}^{N-1} 3\|\phi_h^{n+1} - 2\phi_h^n + \phi_h^{n-1}\|^2 + \nu\Delta t \sum_{n=1}^{N-1} \|\nabla I[\phi_h^{n+1}]\|^2$$
$$(A.28) \quad \leq C\Big(\|u(t_1) - u_h^1\|^2 + \|(u(t_0) - u_h^0)\|^2 + h^{2k+2}\|\|u\|\|_{\infty,k+1}^2$$
$$+ h^{2k+2}\|u_t\|_{2,k+1}^2 + h^{2k}\|\|u\|\|_{2,k+1}^2 + h^{4k}\|\|u\|\|_{4,k+1}^4 + h^{2s+2}\|\|p\|\|_{2,s+1}^2$$
$$+ \Delta t^4(\|u_{ttt}\|_{2,0}^2 + \|\nabla u_{tt}\|_{2,0}^2)\Big).$$

We add both sides of $(A.28)$ with

$$\|\eta^N\|^2 + \|2\eta^N - \eta^{N-1}\|^2 + \|\eta^N - \eta^{N-1}\|^2 + \sum_{n=1}^{N-1} 3\|\eta^{n+1} - 2\eta^n + \eta^{n-1}\|^2$$
$$(A.29)$$
$$+ \nu\Delta t \sum_{n=1}^{N-1} \|\nabla(\frac{3}{2}\eta^{n+1} - \eta^n + \frac{1}{2}\eta^{n-1})\|^2.$$

and apply triangle inequality to get (21). $\qquad\qquad\qquad\qquad\qquad\qquad\square$

Appendix B. Second Order Error Estimator

This section justifies the use of EST_2 as an error estimator for the second order approximation. A Taylor series calculation shows that the second order approximation $y_{(2)}^{n+1}$ in Algorithm 2.1 has the local truncation error (LTE) (for constant stepsize)

$$LTE = -\Delta t^3 \left(\frac{1}{3} y''' + \frac{1}{2} f_y y'' \right) + \mathcal{O}(\Delta t^4).$$

Consider the addition of a second time filter,

$$
\begin{array}{llc}
& \text{Step 1} \quad : & \frac{y_{n+1}^{(1)} - y^n}{\Delta t} = f(t_{n+1}, y_{(1)}^{n+1}), \\
\text{(A.30)} & \text{Step 2} \quad : & y_{(2)}^{n+1} = y_{(1)}^{n+1} - \frac{1}{3}\left\{ y_{(1)}^{n+1} - 2y^n + y^{n-1} \right\} \\
& \text{Step 3} \quad : & y_{n+1} = y_{(2)}^{n+1} - \frac{2}{11}\left\{ y_{(2)}^{n+1} - 3y^n + 3y^{n-1} - y^{n-2} \right\}.
\end{array}
$$

Another Taylor series calculation shows that the induced method has the LTE of

$$LTE = -\Delta t^3 \frac{1}{2} f_y y'' + \mathcal{O}(\Delta t^4),$$

thus, y_{n+1} yields a more accurate (still second order) approximation, and

$$EST_2 = y_{(2)}^{n+1} - y_{n+1} = \frac{2}{11}\left\{ y_{n+1}^{(2)} - 3y^n + 3y^{n-1} - y^{n-2} \right\}$$

gives an estimate for the error of y_{n+1}. This is extended to variable stepsize using Newton interpolation, and written with stepsize ratios in Algorithm 2.1.

This is a nonstandard approach since one would normally use a higher order approximation to estimate the error. However, this is simple since it requires no additional function evaluations or Jacobians, and does not require solving a system of equations. Interestingly, (A.30) remains energy stable, and could be useful as a standalone constant stepsize method.

Department of Mathematics, University of Pittsburgh, Pittsburgh, PA 15260, USA.
E-mail: decariavp@ornl.gov, wjl@pitt.edu and haz50@pitt.edu