5-2023

Advancements in Fluid Simulation Through Enhanced Conservation Schemes

Sean Ingimarson
singima@clemson.edu

Advancements in fluid simulation through enhanced conservation schemes

A Dissertation
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
Mathematics

by
Sean Ingimarson
May 2023

Accepted by:
Dr. Leo Rebholz, Committee Chair
Dr. Timo Heister, Committee Co-Chair
Dr. Hyesuk Lee
Dr. Qingshan Chen

Abstract

To better understand and solve problems involving the natural phenomenon of fluid and air flows, one must understand the Navier-Stokes equations. Branching several different fields including engineering, chemistry, physics, etc., these are among the most important equations in mathematics. However, these equations do not have analytic solutions save for trivial solutions. Hence researchers have striven to make advancements in varieties of numerical models and simulations. With many variations of numerical models of the Navier-Stokes equations, many lose important physical meaningfulness. In particular, many finite element schemes do not conserve energy, momentum, or angular momentum. In this thesis, we will study new methods in solving the Navier-Stokes equations using models which have enhanced conservation qualities, in particular, the energy, momentum, and angular momentum conserving (EMAC) scheme.

The EMAC scheme has gained popularity in the mathematics community over the past few years as a desirable method to model fluid flow. It has been proven to conserve energy, momentum, angular momentum, helicity, and others. EMAC has also been shown to perform better and maintain accuracy over long periods of time compared to other schemes. We investigate a fully discrete error analysis of EMAC and SKEW. We show that a problematic dependency on the Reynolds number is present in the analysis for SKEW, but not in EMAC under certain conditions. To further explore this concept, we include some numerical experiments designed to highlight these differences in the error analysis. Additionally, we include other projection methods to measure performance.

Following this, we introduce a new EMAC variant which applies a differential spatial filter to the EMAC scheme, named EMAC-Reg. Standard models, including EMAC, require especially fine meshes with high Reynold's numbers. This is problematic because the linear systems for 3D flows will be far too large and take an extraordinary amount of time to compute. EMAC-Reg not only performs better on a coarser mesh, but maintains conservation properties as well.

Another topic in fluid flow computing that has been gaining recognition is reduced order models. This method uses experimental data to create new models of reduced computational complexity. We introduce the concept of consistency between a full order and a reduced order model, i.e., using the same numerical scheme for the full order and reduced order model. For inconsistency, one could use SKEW in the full order model and then EMAC for the reduced order model. We explore the repercussions of having inconsistency between these two models analytically and experimentally.

To obtain a proper linear system from the Navier-Stokes equations, we must solve the nonlinear problem first. We will explore a method used to reduce iteration counts of nonlinear problems, known as Anderson acceleration. We will discuss how we implemented this using the finite element library deal.II [5], measure the iteration counts and time, and compare against Newton and Picard iterations.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The importance of accurately and efficiently computing fluid flow phenomena cannot be understated and has exceptionally far reach in many fields. It is useful for pipe and channel flow simulation, car and airplane design, weather prediction, blood flow simulation, ocean currents, astrophysics, and much more. It is also a field with many unanswered questions and room for significant computational improvement.

Despite its importance, it is not practical to physically simulate benchmark fluid flow problems. Consider an experiment where we push water from one end of a channel to the other, with a cylinder placed as an obstruction to the water. One may think that this is a simple task at first, however there is a great deal of engineering that must go into creating this setting [1]. The volume of water that is pushed must be very precise and the velocity needs to be carefully monitored. The machines required for this would also be rather expensive. Another example is an acoustic wind tunnel [50], which is very expensive to make, is not commonly accessible by most academic researchers, and can have trouble simulating many practical flows.

For these reasons, it motivates us to seek cheaper and more opportune methods of understanding fluid flow. Performing numerical simulations is a great solution to this issue. Having software capable of computing solutions to such problems grants more researchers access to knowledge of fluid flow phenomena, among other areas as well.

Over the last few decades, many advancements have been made in fluid flow computation. Researchers have translated well established theory of physical phenomenon into robust numerical methods for problem solving, e.g., finite element and finite volume methods. Many of these methods, however, do not maintain conservation of physical properties such as energy or momentum. Losing numerical conservation of physically conserved properties leads to physical inaccuracies, directly translating into numerical inaccuracies.

In this dissertation, we focus on the Navier-Stokes equations, the equations which determine incompressible fluid flow. We will study the finite element analysis of this equation and create a new physical quantity conserving method method that maintains accuracy. We will also study established quantity conserving methods and how they perform against others. To that end, we will provide several error analyses and numerical experiments. We will introduce new concepts of mixing model types with reduced order models and show implications of poorly chosen combinations. Finally, we integrate a technique of accelerating the fixed point scheme into well-known software.

1.1 Navier-Stokes equations

We study the equations that govern the evolution of incompressible Newtonian fluid flow, which is modeled by the Navier-Stokes equations (NSE). We let u be the fluid velocity, p the fluid pressure, ν the viscosity, and f the external force. The NSE read as follows:

$$\begin{aligned} u_t + u \cdot \nabla u - \nu \Delta u + \nabla p &= f, \\ \nabla \cdot u &= 0. \end{aligned} \tag{1.1}$$

The first equation in (1.1) is the momentum equation and the second is the incompressibility constraint (also known as the conservation of mass equation). This system is what is widely used to simulate most Newtonian fluids including water, ethanol, oil, and air (velocity below 220 m.p.h.). Analytically solving nonlinear partial differential equations (PDEs) with several terms is impractical. Moreover, the theory behind the NSE remains incomplete even in the simplest mathematical case of periodic boundary conditions: it is unknown if there exist strong solutions in three dimensions. It was shown that weak solutions exist, but there is yet to be a result about uniqueness. This is a very famous problem, in fact it is one of the Millenium Prize Problems [30]. Thus, finding analytical solutions to the NSE seems far out of reach.

Another major issue pertains to the Reynolds number (Re), the ratio of inertial forces to viscous forces, which is defined by

$$Re = \frac{\text{inertial forces}}{\text{viscous forces}} = \frac{UL}{\nu}.$$

Here, U is a characteristic flow speed of the fluid (such as the average), L is a characteristic length scale, and ν is the kinematic viscosity. For large values of Re we must have a very fine mesh to obtain an accurate solution, which can be very computationally expensive. Kolmogorov shows in [58, 59, 57, 33] in 1941, that the smallest active scales in a flow are $O(Re^{-3/4})$. This implies that fully resolving 3D flows require mesh widths to be $\Delta x = \Delta y = \Delta z = O(Re^{-3/4})$, which means the total number of meshpoints must be $O(Re^{9/4})$. Industry often simulates flows with Reynolds numbers that are routinely $Re = 10^6$ (for a compact car at 60 mph [61]) but often higher, and so it becomes evident that such simulations can be very computationally costly. In some cases, it is not even possible since needing to solve the resulting linear systems at each time step requires both vast computational resources and the ability to wait for weeks or more for a simulation to finish.

Despite the incompleteness of the theory and the computational shortcomings, there have been major advancements in improving long-term accuracy of NSE simulations. In particular, we will be focusing on an alternative discrete formulation of the NSE. As it stands, the NSE conserves quantities such as kinetic energy, linear momentum, and angular momentum. However, in the discrete setting, we are not guaranteed conservation of these quantities because of the weak enforcement of the divergence free condition. Without such conservation, a model will be physically inaccurate and the numerical results will be impaired. The formulations we study provide multiple appropriate conservation laws in the discrete setting as well as apply spatial filters to improve accuracy on coarser meshes. This will relax requirements on mesh resolution from high values of Re.

1.1.1 Reynolds number

As mentioned above, the Reynolds number is classically defined as the ratio of the inertial forces and the viscous forces. One could also redefine the Reynolds number in terms of the dynamic viscosity, $\mu = \rho\nu$, where ρ is the fluid density. The Reynolds number can even be defined using several different parameters, but we will use this definition.

The Reynolds number, named after Osbourne Reynolds, was originally formulated by George Stokes in 1851 [86], although not named as such until Arnold Sommerfeld coined it [84]. The Reynolds number is useful for predicting the behavior of fluids because it gives insight on the viscosity of the fluid and velocity of the flow. Smaller Reynolds numbers are associated with laminar flow, while higher Reynolds number are associated with turbulent flow. We mainly use moderate Reynolds numbers, such as $Re = 600$ and 1000.

1.1.2 NSE conservation laws

With zero viscosity and no external forcing ($\nu = 0,\ f = 0$), we obtain energy ($E = \frac{1}{2}\int_\Omega |u|^2\, dx$), momentum ($M = \int_\Omega u\, dx$), and angular momentum ($AM = \int_\Omega (u \times x)\, dx$) conservation in the NSE. We use the standard $L^2(\Omega)$ norm $||\cdot||$ and inner product $(\cdot,\cdot)$. We now show this to be true on the continuous level assuming that $u = 0$ on a strip along $\partial\Omega$.

1.1.2.1 Energy

We define energy as $E = \frac{1}{2}(u,u) = \frac{1}{2}\int_\Omega |u|^2\, dx$. In order to recover this term from the NSE, we multiply the momentum equation by u, integrate, and apply Stokes theorem. This gives

$$(u_t, u) + (u \cdot \nabla u, u) + (\nabla p, u) + \nu\, ||\nabla u||^2 = (f, u)\,.$$

For the pressure term, we integrate by parts and apply the divergence theorem to get

$$(\nabla p, u) = -(p, \nabla \cdot u) - \int_{\partial\Omega} p(u \cdot n)\, dx.$$

By the conservation of mass equation, the first term on the right hand side is zero. Additionally, our assumption that u vanishes on the boundary gives us that the other term on the right hand side is zero, hence $(\nabla p, u) = 0$. Then for the nonlinear term, we have

$$(u \cdot \nabla u, u) = -\frac{1}{2}\left((\nabla \cdot u)u, u\right) = 0.$$

Our assumption of $\nu = 0$ and $f = 0$ now leaves us with

$$(u_t, u) = 0.$$

A simple identity now provides

$$\frac{1}{2}\frac{d}{dt}\left|\left|u\right|\right|^2 = 0,$$

which is the desired result.

1.1.2.2 Momentum

For momentum conservation, we multiply the unit vector e_i, where $i = 1, 2, 3$, to the NSE momentum equation and integrate. Then using Stokes theorem, we get

$$\left(u_t, e_i\right) + \left(u \cdot \nabla u, e_i\right) + \left(\nabla p, e_i\right) + \nu\left(\nabla u, \nabla e_i\right) = \left(f, e_i\right).$$

For the nonlinear term, we use a common trilinear term identity,

$$\left(u \cdot \nabla u, e_i\right) = -\left(u \cdot \nabla e_i, u\right) - \left((\nabla \cdot u)u, e_i\right) = 0.$$

The first term on the right hand side disappears because $\nabla e_i = 0$ and the second term disappears due to the conservation of mass equation. Then $\left(\nabla p, e_i\right) = \left(p, \nabla \cdot e_i\right) = 0$ (using the divergence theorem and the fact that $p = 0$ on $\partial\Omega$). This leaves us with

$$\frac{d}{dt}\int_{\Omega} u \, dx = 0,$$

which is exactly momentum conservation.

1.1.2.3 Angular momentum

Define $\phi_i = x \times e_i$ with $i = 1, 2, 3$. Now multiply the NSE by ϕ_i, integrate over Ω, and use Stokes theorem to get

$$\left(u_t, \phi_i\right) + \left(u \cdot \nabla u, \phi_i\right) - \left(p, \nabla \cdot \phi_i\right) + \nu\left(\nabla u, \nabla \phi_i\right) = \left(f, \phi_i\right).$$

For the trilinear term, we use a similar technique as for momentum conservation,

$$\left(u \cdot \nabla u, \phi_i\right) = -\left(u \cdot \nabla \phi_i, u\right) - \left((\nabla \cdot u)u, \phi_i\right).$$

The second term on the right hand side is zero by the conservation of mass equation, so we must investigate the first term, specifically $\nabla\phi_i$. It can be easily verified that $\nabla\phi_i$ is skew-symmetric for $i = 1, 2, 3$, which gives

$$(u \cdot \nabla\phi_i, u) = 0.$$

Then for the pressure term, we have that $\nabla \cdot \phi_i = 0$ for $i = 1, 2, 3$. Using $\nu = 0$ and $f = 0$ leaves us with

$$0 = (u_t, \phi_i) = \int_\Omega u_t \cdot (\vec{x} \times e_i),$$

and so $\frac{d}{dt}\int_\Omega (u \times \vec{x}) \cdot e_i = 0$ for $i = 1, 2, 3$.

1.2 Conservation issues with standard schemes

The standard FEM discretization scheme for the NSE is not the most robust model in many scenarios. This might be a strange claim, but when we discretize models, we lose important relations needed for conservation. For example, the standard convective scheme loses energy conservation upon discretization. This is a direct result from losing the strong definition of incompressibility, which will be explored in more detail. In this section, we provide some standard semi-discrete FEM schemes for the NSE and discuss conservation issues.

We define the following formulations of the NSE:
Find $(u_h, p_h) \in (X_h, Q_h)$ such that for every $(v_h, q_h) \in (X_h, Q_h)$,
Convective formulation (CONV):

$$\begin{aligned} ((u_h)_t, v_h) + (u_h \cdot \nabla u_h, v_h) - (p, \nabla \cdot v_h) + \nu (\nabla u_h, \nabla v_h) &= (f, v_h), \\ (\nabla \cdot u_h, v_h) &= 0. \end{aligned}$$

Skew-symmetric formulation (SKEW):

$$
\begin{aligned}
((u_h)_t, v_h) + (u_h \cdot \nabla u_h, v_h) + \frac{1}{2}((\nabla \cdot u_h)u_h, v_h) - (p, \nabla \cdot v_h) + \nu(\nabla u_h, \nabla v_h) &= (f, v_h), \\
(\nabla \cdot u_h, v_h) &= 0.
\end{aligned}
$$

These formulations have been utilized for several decades [89] and have had thousands of numerical tests performed using them. Additionally, the analysis done for them is very refined. However, these formulations lack important quantity conservation. We have shown previously that the strong form of the NSE conserves energy, momentum, and angular momentum, but that is not necessarily true when we discretize [19]. This is mainly due to the fact that we are not guaranteed $\nabla \cdot u_h = 0$ except in very special circumstances such as using Scott-Vogelius or other divergence-free element pairs. Without this assumption, neither CONV or SKEW conserve all three of these quantities.

To address this issue of discretizations not preserving important physical balances, the EMAC scheme was developed for the NSE in [19], we define it here:
Find $(u_h, p_h) \in (X_h, Q_h)$ such that for every $(v_h, q_h) \in (X_h, Q_h)$, Energy, momentum, and angular momentum conserving formulation (EMAC):

$$
\begin{aligned}
((u_h)_t, v_h) + (2D(u_h)u_h, v_h) + ((\nabla \cdot u_h)u_h, v_h) - (p, \nabla \cdot v_h) + \nu(\nabla u_h, \nabla v_h) &= (f, v_h), \\
(\nabla \cdot u_h, v_h) &= 0.
\end{aligned}
$$

This formulation conserves energy, momentum, and angular momentum by design even when the divergence constraint is not strongly enforced (which is the typical case with finite element and finite difference methods). It is the first such scheme to conserve these quantities in typical finite element discretizations where the divergence constraint is weakly enforced. We note that if newly developed pointwise divergence-free finite elements are used, e.g. [6, 98, 29, 43, 42], then the numerical velocity found with EMAC will be the same as recovered from more traditional formulations such as CONV and SKEW, and all of them will conserve energy, momentum, and angular momentum. However, the development of these strongly divergence-free methods is still quite new. It typically requires non-standard meshing and elements and is not yet included into major finite element software packages such as deal.II [5].

Although EMAC has been analyzed and tested [51, 53, 52, 68, 20, 10, 64], it has not been given nearly as much treatment as more well-known formulations. We believe it critical to further extend our knowledge of EMAC theoretically and computationally. It has been shown that the semi-discretization in time using Crank-Nicolson timestepping of EMAC has second order convergence as well as a much stronger bound than SKEW with respect to Re [68]. We take this one step further and give an error analysis for the full discretization.

1.3 Filtered equations and LES models

We mention previously that computing with standard models can require extremely fine meshes. To address this problem, many models have been developed that approximate the NSE, but can be solved on much coarser meshes than the NSE requires. Some common examples are $k-\epsilon$ type models [71], Smagorinsky type models [83, 9], and our interest herein is a model that fits in the class of Large Eddy Simulation (LES) models [9, 63, 76, 27]. LES models aim to accurately estimate the large scales of the flow and model the effect of small scales on the large scales. Hence the goal of LES is not pointwise accurate solutions, but instead solutions that agree with NSE on general or averaged flow behavior. Regularization models are our particular interest herein, and these are LES models that use filtering/averaging operations applied to the NSE to reduce the complexity of the system by eliminating finer scales and steepening the slope of the energy cascade at scales smaller than can be resolved on a given mesh. NS-α and Leray-α (and their many variants), for example, are popular regularization models that have been extensively studied in recent years. They are found to have many desirable mathematical properties (e.g. well-posedness, fidelity to various physical balances [31, 32, 74, 23, 36, 63, 25, 95]) and be successful in simulating high Reynolds number and turbulent flows [48, 23, 37, 21, 22, 75, 26, 11].

Despite the attractive analytical properties that regularization models enjoy, many of these properties can be lost in a discretization, in particular the balances of physical quantities. In fact, this is also true for discretizations of the NSE, which after being discretized generally lose the exact balances of most (if not all) of energy, momentum, angular momentum, helicity, and others. The situation for discretized regularization models is worse, since the modeling process already removes or alters some important physical balances, and then the discretization process exacerbates the problem. Of course, in any model of the NSE one must sacrifice some physical accuracy, since one is no longer solving the NSE. However, the goal for simulations should be to maintain as much physical accuracy as possible, so that solutions are physically meaningful and can be confidently used as surrogates for the true physical model.

One of our goals is to extend the EMAC technology to a regularization model for the NSE. The model that we discretize takes the form

$$u_t + w \cdot \nabla w + \nabla p - \nu \Delta u = f, \tag{1.2}$$

$$-\alpha^2 \Delta w + w = u, \tag{1.3}$$

$$\nabla \cdot w = 0. \tag{1.4}$$

Here, (1.3) uses a fourth order approximation to the Gaussian filter with filtering radius α. Under periodic boundary conditions, $\nabla \cdot w = 0 \implies \nabla \cdot u = 0$ since the filter operation commutes with the divergence operator. Also for smooth solutions to (1.2)-(1.4), we may apply the divergence operator to (1.3) and recover $\nabla \cdot u = 0$. For weak solutions that are not strong, if one wishes to enforce additionally that $\nabla \cdot u = 0$, then one may add a Lagrange multiplier (i.e. pressure) term to (1.3); however, we assume throughout that strong solutions to (1.2)-(1.4) exist under the assumption of homogeneous Dirichlet boundary conditions for u and w. The following formulations are examples of utilizing this spatial filter after discretization.

Find $(u_h, p_h, w_h, \lambda_h) \in (X_h, Q_h, X_h, Q_h)$ such that for every (v_h, q_h, χ_h, r_h),

NS-α formulation

$$\begin{aligned} ((u_h)_t, v_h) + (u_h \times w_h, v_h) - (p, \nabla \cdot v_h) + \nu (\nabla u_h, \nabla v_h) &= (f, v_h), \\ (\nabla \cdot u_h, v_h) &= 0, \\ (\lambda_h, \nabla \cdot \chi_h) + \alpha^2 (\nabla w_h, \nabla \chi_h) + (w_h, \chi_h) &= (u_h, \chi_h), \\ (\nabla \cdot w_h, r_h) &= 0. \end{aligned}$$

Leray-α formulation

$$\begin{aligned} ((u_h)_t, v_h) + (w_h \times u_h, v_h) - (p, \nabla \cdot v_h) + \nu (\nabla u_h, \nabla v_h) &= (f, v_h), \\ (\nabla \cdot u_h, v_h) &= 0, \\ (\lambda_h, \nabla \cdot \chi_h) + \alpha^2 (\nabla w_h, \nabla \chi_h) + (w_h, \chi_h) &= (u_h, \chi_h), \\ (\nabla \cdot w_h, r_h) &= 0. \end{aligned}$$

The specific formulation for the regularization model (1.2)-(1.4) is chosen so that it fits the form of a model that the nonlinearity formulation $2D(w)w + (\nabla \cdot w)w$ (which is identical to $w \cdot \nabla w$ when $\nabla \cdot w = 0$, up to a potential term) will preserve energy, momentum, and angular momentum when $\nabla \cdot w \neq 0$. We note that the nonlinear forms $w \cdot \nabla u$ (i.e. that of Leray-α) or $(\nabla \times u) \times w$ (i.e. that of NS-α) need not preserve each of energy, momentum, and angular momentum when pointwise divergence free for velocities and/or filtered velocities (see appendix A). In other words, (1.2)-(1.4) is the α regularization model that naturally fits the EMAC framework.

Herein, we study and test discretizations of (1.2)-(1.4) as well as show that it conserves energy, momentum, and angular momentum; we denote it as the Energy, Momentum, and Angular Momentum conserving regularization formulation (EMAC-Reg). We formally define this scheme in Chapter 4, followed by showing it is stable, well-posed, the aforementioned quantities are conserved, and is optimally accurate. Following this, we will give an error analysis of the fully discrete EMAC formulation.

1.4 Reduced Order Models

Reduced order models (ROMs) are computational models whose computational complexity is magnitudes lower than that of a standard full order model (FOM) obtained from a standard numerical discretization method. As mentioned above, full numerical FEM schemes have a high computational cost. ROMs are constructed with computational data that has already been calculated, so it cuts out excessive computation time. One could classify or group ROMs in a myriad of ways:

Nonintrusive ROMs: Examples include machine learning ROMS [2] and sparse identification of nonlinear dynamics [15].

Intrusive ROMs: Examples include Galerkin ROMs (G-ROMs) and Petrov-Galerkin ROMs (PG-ROMs) [17]. More specifically, examples of G-ROMs include the proper orthogonal decomposition (POD) [49] and reduced basis method (RBS) [45].

Using one type of ROM has advantages and disadvantages over using another. Intrusive ROMs require more details about the data than nonintrusive ROMs, so they are not as simple to implement. Alternatively, if one were to prioritize the framework on which the ROM is built, then the intrusive ROM is more advantageous. Particularly, the G-ROM is constructed on the standard Galerkin framework. This allows us to prove fundamental properties such as stability, consistency, and convergence (proven in [60, 82, 56] for POD and [45] for RBS).

We will address questions on the mathematical foundations for G-ROMs. Mainly: If we construct both FOMs and ROMs on a Galerkin foundation, should we use the same discretization scheme? Which will yield more accurate ROMs? We will investigate these questions with our new definition for FOM-ROM consistency.

Definition 1 (FOM-ROM Consistency)**.** *A ROM is FOM consistent if it uses the same computational model and the same numerical discretization (e.g., spatial discretization, time discretization, and nonlinearity and pressure treatment) as the FOM.*

In Chapter 5, we show the analytical implications of a ROM being FOM-ROM inconsistent. We will test models that are FOM-ROM consistent and those that are FOM-ROM inconsistent and compare common quantities found in benchmark numerical tests.

1.5 Anderson acceleration

Anderson acceleration (AA) is a strategy which speeds up convergence of fixed point iterations, an idea originally manufactured by D.G. Anderson in 1965 [4]. It has been useful in several different disciplines such as flow problems [66, 70, 35], radiation diffusion and nuclear physics [3, 92], machine learning [34], linear solves [72, 73], convex optimization [67], and robotics [69]. In its original work, AA was applied to nonlinear integral equations that stem from research in the kinetic theory of gases. However, it was in [93] where AA was put in the spotlight and started to pick up in the mathematics community. This is relevant to this thesis because nonlinear solvers are needed to solve the nonlinear problems herein. Of course, AA is useful far beyond these PDEs.

Starting with standard fixed point theory, we start with establishing the nonlinear contractive operator.

Definition 2. *For a normed space X and norm $||\cdot||_X$, $g : X \to X$ is a contractive operator with contraction ratio $r \in (0,1)$ if*

$$||g(x) - g(y)||_X \leq r\,||x - y||_X\,,\ \textit{for all}\ x, y \in X.$$

It is known that if g is a contractive operator following definition 2, there exists a unique $u^* \in X$ such that $g(u^*) = u^*$. Then we have the actual definition for the fixed point iteration.

Algorithm 3 (Fixed point iteration)**.** *Given space X and contractive function g,*

```
Step 0: Choose $x_0 \in X$.
Step 1: For $k = 1, 2, \ldots$
        $x_{k+1} = g(x_k)$.
```

With this we can establish the AA algorithm. The algorithm that we provide is generalized, and we will choose a specific function for g and X later in the paper. However, for this definition we must assume that X is a Hilbert space.

Algorithm 4 (Anderson acceleration)**.** *Let X be a Hilbert space and $g : X \to X$ be a contractive function. The Anderson acceleration with depth m is as follows:*

```
Step 0: Choose $u_0 \in X$.
Step 1: Find $\tilde{u}_1 \in X$ such that $\tilde{u}_1 = g(u_0)$. Set $u_1 = \tilde{u}_1$.
Step k: For $k = 1, 2, \ldots$
  a.) Find $\tilde{u}_{k+1} = g(u_k)$.
  b.) Solve the minimization problem for $\{\alpha_j^{k+1}\}_{k-m}^{k}$
        $\min_{\sum_{j=k-m}^{k} \alpha_j^{k+1} = 1} \left\| \sum_{j=k-m}^{k} \alpha_j^{k+1} (\tilde{u}_{j+1} - u_j) \right\|_X .$
  c.) Set $u_{k+1} = \sum_{j=k-m}^{k} \alpha_j^{k+1} \tilde{u}_{j+1}$.
```

The convergence of AA has been widely studied [28, 91, 85, 97]. It was shown in [28] that convergence is improved for fixed point methods that converge linearly. However, it shows that applying AA slows down convergence for fixed point methods that converge quadratically, e.g., Newton iterations, in agreement with numerical tests in the literature. We will show various results of applying AA to different problems in deal.II [5] later. deal.II is an open source finite element library for C++ which is maintained by authors at Oak Ridge National Laboratory, Colorado State University, Clemson University, and others. We will apply different implementations of AA, i.e., an AA code we created and applied to the deal.II tutorial step-57 and another from nonlinear solver library KINSol [88].

1.6 Structure of dissertation

We focus on showing a fully discretized error analysis of EMAC as well as introducing the EMAC-Reg formulation and demonstrating its strengths on coarser meshes. In Chapter 2 we will provide the necessary notation and preliminary information used in further chapters. In Chapter 3 we study the convergence of SKEW and EMAC with Crank-Nicolson timestepping with different assumptions on the smoothness of velocity. In Chapter 4 we introduce a semi-discretized EMAC-Reg formulation and show that it is a well-posed problem. Additionally, we show that it conserves energy, momentum, and angular momentum just as EMAC does. We also give numerical experiments which compare EMAC-Reg to other formulations on a coarse mesh. Chapter 5 contains an error analysis on a FOM-ROM inconsistent system and numerical results to reinforce the theory. With Chapter 6, we show results from various implementations of Anderson acceleration in deal.II. While this last Chapter may seem somewhat disjoint, EMAC requires solving nonlinear problems at each time step in order to maintain conservation properties [20]. Hence methods such as AA are needed for efficient EMAC simulations.

Chapter 2

Notation and preliminaries

We consider a convex polygonal domain $\Omega \subset \mathbb{R}^d$, $d = 2, 3$. Denote the $L^2(\Omega)$ inner product and norm on Ω by $(\cdot, \cdot)$ and $\| \cdot \|$, respectively, and we note all other norms will be labeled with subscripts. The natural velocity and pressure spaces are

$$X = \left\{ v \in H^1(\Omega)^d,\ v|_{\partial\Omega} = 0 \right\}, \quad Q = \left\{ q \in L^2(\Omega),\ \int_\Omega q dx = 0 \right\}.$$

Let V denote the divergence-free subspace of X, $V := \{v \in X \ : \ (\nabla \cdot v, q) \ = 0,\ \forall q \in Q\}$. We also define the dual of X and its norm,

$$X' = H^{-1}(\Omega), \quad ||f||_{X'} = \sup_{v \in X'} \frac{(f, v)}{||v||_X},$$

for any $f \in L^2(\Omega)$.

We further consider subspaces $X_h \subset X$, $Q_h \subset Q$ to be finite element velocity and pressure spaces corresponding to a conforming triangulation $\mathcal{T}_h$ of Ω, where h is the global mesh-size. For $\mathcal{T}_h$ we assume the minimal angle condition if h varies. Define the discretely divergence-free subspace of X_h by

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

Most common finite element (FE) discretizations of the NSE and related systems enforce the divergence constraint $\nabla \cdot u_h \ = 0$ weakly (and thus $V_h \not\subset V$ except in special cases) and use Taylor-Hood (P_k, P_{k+1})

elements [61].

Define the operator A_h by: Given $\phi \in H^1(\Omega)$, $A_h\phi \in V_h$ solves

$$(A_h\phi, v_h) \,=-\, (\nabla\phi, \nabla v_h) \tag{2.1}$$

for all $v_h \in V_h$.

We denote I_h^{St} as the discrete Stokes projection operator [80], which is defined by: Given $\phi \in V$, find $I_h^{St}\phi \in V_h$ satisfying

$$(\nabla I_h^{St}\phi, \nabla v_h) = (\nabla\phi, \nabla v_h), \quad \forall v_h \in V_h. \tag{2.2}$$

Following [80], we assume $X_h = X \cap P_k(\mathcal{T}_h)$ and $Q_h = Q \cap P_{k-1}(\mathcal{T}_h)$, with P_k being the set of degree $k \geq 2$ piecewise polynomials over triangles. Additionally, we assume that these spaces satisfy the infsup condition [38]. We make use of the following approximation properties [14]:

$$\begin{aligned}
\inf_{v \in X_h} ||u - v||_{L^2} &\leq Ch^{k+1}\, ||u||_{k+1}\,, \\
\inf_{v \in X_h} ||u - v||_{H^1} &\leq Ch^{k}\, ||u||_{k+1}\,, \\
\inf_{r \in Q_h} ||p - r||_{L^2} &\leq Ch^{s+1}\, ||p||_{s+1}\,,
\end{aligned} \tag{2.3}$$

where $u \in H^{k+1}(\Omega)$ and $p \in H^{s+1}(\Omega)$. Additionally, the approximation properties of the Stokes projection operator in L^2 and H^1 norms follow from the interpolation properties of the finite element spaces [38]:

$$\left|\left|\nabla I_h^{St}\phi\right|\right|_{L^r} \leq C\, ||\nabla\phi||_{L^r}\,, \quad r \in [2, \infty]. \tag{2.4}$$

The analysis we show later utilizes a discrete Gronwall inequality [46, 61], given by the following lemma.

Lemma 5. *Let Δt, H, and a_n, b_n, c_n, γ_n (for integers $n \geq 0$) be nonnegative numbers such that*

$$a_l \,+\, \Delta t \sum_{n=0}^{l} b_n \;\leq\; \Delta t \sum_{n=0}^{l} \gamma_n\, a_n \;+\; \Delta t \sum_{n=0}^{l} c_n \;+\; H \quad \textit{for } l \geq 0\,. \tag{2.5}$$

Suppose that $\Delta t\,\gamma_n < 1$, for all n. Then,

$$a_l + \Delta t \sum_{n=0}^{l} b_n \;\le\; \exp\left(\Delta t \sum_{n=0}^{l} \frac{\gamma_n}{(1-\Delta t\,\gamma_n)} \right) \left\{ \Delta t \sum_{n=0}^{l} c_n + H \right\} \quad \textit{for } l \ge 0\,.$$

In particular, we will be using the next version of the Gronwall inequality. It's given as a remark to Lemma 5.1 in [46] and is as follows:

Lemma 6. *Let Δt, H, and a_n, b_n, c_n, d_n (for integers $n \ge 0$) be finite nonnegative numbers such that*

$$a_l + \Delta t \sum_{n=0}^{l} b_n \le \Delta t \sum_{n=0}^{l-1} d_n a_n + \Delta t \sum_{n=0}^{l} c_n + H \quad for\ l \ge 1.$$

Then, for $\Delta t > 0$

$$a_l + \Delta t \sum_{n=0}^{l} b_n \le \exp\left(\Delta t \sum_{n=0}^{l-1} d_n \right) \left(\Delta t \sum_{n=0}^{l} c_n + H \right) \quad for\ l \ge 1.$$

2.1 Vector identities and trilinear forms

The EMAC formulation uses the identity

$$(u \cdot \nabla) u = 2D(u)u - \frac{1}{2}\nabla |u|^2, \tag{2.6}$$

where u is a sufficiently smooth gradient field and $D(u) = \frac{1}{2}(\nabla u + (\nabla u)^T)$ is the symmetric part of the gradient. This identity splits the inertia term into the acceleration driven by $2D(u)$ and potential term that is absorbed by redefined pressure (defined in Theorem 3.4). Based on (2.6) the trilinear form for EMAC (Galerkin) formulation is defined by

$$c(u, v, w) = 2(D(u)v, w) + ((\nabla \cdot u)v, w).$$

We also define the nonlinear term for SKEW,

$$b^*(u, v, w) = (u \cdot \nabla v, w) + \frac{1}{2}\left((\nabla \cdot u)v, w\right).$$

Herein, we assume $u,\ v,\ w \in X$ (no divergence free condition is assumed for any of $u,\ v,\ w$). The trilinear form c was developed for EMAC in [19] as a consistent weak representation of the NSE nonlinearity that preserves energy, momentum, and angular momentum. To accommodate for the change in nonlinear term, we redefine the pressure term,

$$P = p - \frac{1}{2}|u|^2.$$

We will also be utilizing these identities,

$$\begin{aligned}(u \cdot \nabla v, w) &= -(u \cdot \nabla w, v) - ((\nabla \cdot u)v, w), &(2.7)\\ (u \cdot \nabla w, w) &= -\frac{1}{2}((\nabla \cdot u)w, w), &(2.8)\\ (u \cdot \nabla v, w) &= ((\nabla v)u, w) = \left((\nabla v)^T w, u\right). &(2.9)\end{aligned}$$

2.2 True solutions and fully-discretized formulations using SKEW and EMAC

We define the the true solution of the NSE at time t^{n+1} using the SKEW nonlinear term, denoted b^*. Assume u is sufficiently smooth and homogeneous Dirichlet boundary conditions. The true solution u and p at time t^{n+1} satisfy,

$$\begin{aligned}\left(u_t(t^{n+1}), v_h\right) &+ b^*(u(t^{n+1}), u(t^{n+1}), v_h) + \nu\left(\nabla u(t^{n+1}), \nabla v_h\right)\\ &- \left(p(t^{n+1}), \nabla \cdot v_h\right) = \left(f(t^{n+1}), v_h\right), &(2.10)\\ &\left(\nabla \cdot u(t^{n+1}), q_h\right) = 0. &(2.11)\end{aligned}$$

We will also define the true solution of the NSE using the EMAC formulation. The true solution u and P satisfies

$$\begin{aligned}\left(u_t(t^{n+1}), v_h\right) &+ c(u(t^{n+1}), u(t^{n+1}), v_h) + \nu\left(\nabla u(t^{n+1}), \nabla v_h\right)\\ &- \left(P(t^{n+1}), \nabla \cdot v_h\right) = \left(f(t^{n+1}), v_h\right), &(2.12)\\ &\left(\nabla \cdot u(t^{n+1}), q_h\right) = 0. &(2.13)\end{aligned}$$

Here we define the fully discretized formulations using Crank-Nicolson time-stepping with CONV, SKEW, and EMAC nonlinear terms, first we will define CONV. Let $t^{n+\frac{1}{2}} = \frac{t^{n+1}+t^n}{2}$, then find $(u_h^{n+1}, p_h^{n+1}) \in (X_h, Q_h) \times (0, T]$ satisfying for all $(v_h, q_h) \in (X_h, Q_h)$,

$$\begin{aligned} \left(\frac{u_h^{n+1} - u_h^n}{\Delta t}, v_h \right) &+ b(u_h^{n+\frac{1}{2}}, u_h^{n+\frac{1}{2}}, v_h) + \nu \left(\nabla u_h^{n+\frac{1}{2}}, \nabla v_h \right) \\ - \left(p_h^{n+\frac{1}{2}}, \nabla \cdot v_h \right) &= \left(f^{n+\frac{1}{2}}, v_h \right), \qquad (2.14) \\ \left(\nabla \cdot u_h^{n+\frac{1}{2}}, q_h \right) &= 0. \qquad (2.15) \end{aligned}$$

Now we define SKEW. Let $t^{n+\frac{1}{2}} = \frac{t^{n+1}+t^n}{2}$, then find $(u_h^{n+1}, p_h^{n+1}) \in (X_h, Q_h) \times (0, T]$ satisfying for all $(v_h, q_h) \in (X_h, Q_h)$,

$$\begin{aligned} \left(\frac{u_h^{n+1} - u_h^n}{\Delta t}, v_h \right) &+ b^*(u_h^{n+\frac{1}{2}}, u_h^{n+\frac{1}{2}}, v_h) + \nu \left(\nabla u_h^{n+\frac{1}{2}}, \nabla v_h \right) \\ - \left(p_h^{n+\frac{1}{2}}, \nabla \cdot v_h \right) &= \left(f^{n+\frac{1}{2}}, v_h \right), \qquad (2.16) \\ \left(\nabla \cdot u_h^{n+\frac{1}{2}}, q_h \right) &= 0. \qquad (2.17) \end{aligned}$$

Now we have the EMAC formulation with Crank-Nicolson time-stepping: Find $(u_h^{n+1}, P_h^{n+1}) \in (X_h, Q_h) \times (0, T]$ satisfying for all $(v_h, q_h) \in (X_h, Q_h)$,

$$\begin{aligned} \left(\frac{u_h^{n+1} - u_h^n}{\Delta t}, v_h \right) &+ c(u_h^{n+\frac{1}{2}}, u_h^{n+\frac{1}{2}}, v_h) + \nu \left(\nabla u_h^{n+\frac{1}{2}}, \nabla v_h \right) \\ - \left(P_h^{n+\frac{1}{2}}, \nabla \cdot v_h \right) &= \left(f^{n+\frac{1}{2}}, v_h \right), \qquad (2.18) \\ \left(\nabla \cdot u_h^{n+\frac{1}{2}}, q_h \right) &= 0. \qquad (2.19) \end{aligned}$$

There exists extensive literature on the accuracy of (2.14)-(2.15), see e.g., [68, 61, 90], which proves the following for $(X_h, Q_h) = (P_k, P_{k-1})$ Taylor-Hood elements if the SKEW or EMAC forms are used. For a sufficiently smooth NSE solution and sufficiently small step size,

$$\left|\left|u(T) - u_h^M\right|\right|^2 + \nu \Delta t \sum_{n=1}^{M} ||\nabla(u(t^n) - u_h^n)||^2 \leq C(\Delta t^2 + h^{2k}). \qquad (2.20)$$

If BDF2 timestepping is used, then the Δt^2 becomes a Δt^4, and for EMAC the constant on the right hand side does not explicitly depend on Re, while it does for SKEW and other forms [68].

2.3 Notation for the discrete time method

For functions $v(x,t)$ and $1 \leq p \leq \infty$, we define

$$|||v|||_{\infty,k} = \max_{0\leq n\leq M} ||v^n||_k \,, \qquad |||v_{1/2}|||_{\infty,k} = \max_{1\leq n\leq M} \left|\left|v^{n-1/2}\right|\right|_k \,,$$

$$|||v|||_{p,k} = \left(\sum_{n=0}^{M} ||v^n||_k^p \, \Delta t \right)^{\frac{1}{p}} , \qquad |||v_{1/2}|||_{p,k} = \left(\sum_{n=0}^{M} \left|\left|v^{n-1/2}\right|\right|_k^p \Delta t \right)^{\frac{1}{p}} .$$

The error analyses requires such norms that are analogous to the continuous time norms that we are familiar with.

2.4 Reduced order model construction

To create our G-ROM, we start with the snapshot matrix, the matrix which contains the data from the FOM computation. Consider the solution to the fully discretized NSE to our FOM model (2.16)-(2.17). Let $\mathbf{u}_h(t) = \begin{bmatrix} u_h(t) & p_h(t) \end{bmatrix}^T$ be our finite element solution at time t. We construct a matrix whose columns are the coefficient vectors at certain timesteps,

$$U = \begin{bmatrix} \mathbf{u}_h(t_1) & \mathbf{u}_h(t_2) & \dots & \mathbf{u}_h(t_n) \end{bmatrix} ,$$

where n is the number of timesteps. We perform a proper orthogonal decomposition on U using the L^2 inner product, giving the new orthogonal matrix

$$\Phi = \begin{bmatrix} \varphi_1 & \varphi_2 & \dots & \varphi_n \end{bmatrix} ,$$

where φ_i is the ith eigenvector of U^TU. We will use the vectors of Φ to construct our ROM basis [49]. We consider the first r vectors of Φ, the span of this is our new ROM basis, namely

$$V_r = \text{span} \left\{ \varphi_1, \varphi_2, \dots, \varphi_r \right\} .$$

The value we choose for r typically depends on the size of the eigenvalues of $U^T U$. The size of the eigenvalues is proportional to how impactful that basis function is on the velocity. Hence practitioners usually set r equal to the rank of eigenvalue that is the first to be a certain scale smaller than the largest eigenvalue. So if λ_r is the first eigenvalue such that $\lambda_r \leq 10^{-6}\lambda_1$ is true, then r is the dimension of our ROM space.

The next step in the G-ROM construction is the Galerkin step itself, which one can easily guess is very similar to a standard NSE Galerkin scheme such as (2.16)-(2.17). We simply replace u_h with $w_r \in V_r$ and project the entire equation from there onto V_r. We also note that the pressure term will vanish because functions in V_r satisfy the weak condition of mass conservation. Hence, our new G-ROM scheme is as follows: Given $w_r^0 \in V_r$, find $w_r^{n+1} \in V_r$ that satisfies

$$\left(\frac{w_r^{n+1} - w_r^n}{\Delta t}, v_r\right) + b^*(w_r^{n+\frac{1}{2}}, w_r^{n+\frac{1}{2}}, v_r) + \nu\left(\nabla w_r^{n+\frac{1}{2}}, \nabla v_r\right) = \left(f^{n+\frac{1}{2}}, v_r\right), \tag{2.21}$$

for all $v_r \in V_r$. The SKEW nonlinear form in (2.21) can take on any nonlinear form we have mentioned.

To better understand the construction of equation (2.21), we note that w_r^{n+1} is a linear combination of our ROM basis elements, namely

$$w_r^{n+1} = \sum_{j=1}^{r} \alpha_j \varphi_j.$$

If one were to construct a stiffness matrix $S = \left(\nabla w_r^{n+1}, \nabla v_r\right)$, the basic finite element theory gives $S_{i,j} = \left(\varphi_i, \varphi_j\right)$. The main difference between S and a stiffness matrix constructed using standard Galerkin FEM is that S will be dense. The basis elements of V_r are functions that are nonzero over all of Ω, rather than locally nonzero like FEM basis elements. Applying this to the nonlinear term gives

$$b^*(w_r^{n+1}, w_r^{n+1}, v_r) = \sum_{j=1}^{r}\sum_{i=1}^{r} \alpha_i \beta_j b^*(\varphi_i, \varphi_j, v_r).$$

Here, we get a rank 3 tensor, $NL_{i,j,k} = b^*(\varphi_i, \varphi_j, \varphi_k)$. Similar to S, NL is a dense rank 3 tensor. If we choose $r = 9$ for example, we are doing 81 inner products for S and 729 inner products for NL. Although the linear solve at the end is much smaller, the matrix assembly is still rather expensive.

It can also be shown that the POD projection error satisfies the following equality [82]:

$$\frac{1}{M+1}\sum_{j=0}^{M}\left|\left|\nabla\left(u_h(\cdot,t_j)-\sum_{i=1}^{r}(u_h(\cdot,t_j),\psi_i(\cdot))\psi_i(\cdot)\right)\right|\right|^2=\sum_{k=r+1}^{d}||\nabla\psi_k||^2\,\lambda_k, \tag{2.22}$$

where d is the rank of the snapshot matrix, A, and λ_k is the eigenvalue corresponding to the ROM basis function ψ_k.

Chapter 3

An improved analysis for EMAC over longer time intervals

In this chapter, we will give and compare the Crank-Nicolson discretized schemes SKEW (2.16)-(2.17) and EMAC (2.18)-(2.19).

We will perform 2 analyses for both formulations assuming $u^{n+1} \in H^2(\Omega)$ or $H^3(\Omega)$. The error analysis for SKEW is well-documented and considered classic [61]. When we apply the higher restriction of requiring $u^{n+1} \in H^3(\Omega)$, the Gronwall constant of our error term relies less heavily on Re, and in the EMAC case, independent of Re.

3.1 Analysis with $H^2(\Omega)$ regularity assumption on u^{n+1}

Theorem 7. *Let (u_h^{n+1}, p_h^{n+1}) solve (2.16)-(2.17) and (u^{n+1}, p^{n+1}) solve (2.10)-(2.11) be a NSE solution with $u_t^{n+1} \in X'$, $u^{n+1} \in H^2(\Omega)$, and $p^{n+1} \in H^1(\Omega)$, for $0 \leq n \leq M$. Denote $e^n = u^n - u_h^n$, $\eta^n = u^n - I_h^{St}(u^n)$, and $\phi_h^n = I_h^{St}(u^n) - u_h^n$.*

(i) Then for all $0 \leq n \leq M$, the following holds for SKEW

$$\left|\left|e^M\right|\right|^2 + \nu\Delta t \sum_{n=0}^{M-1} \left|\left|\nabla e^{n+\frac{1}{2}}\right|\right|^2 \leq \exp\left(C\Delta t \sum_{n=0}^{M-1}\left(\nu^{-3}\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|^4 + \frac{3}{2}\right)\right) F(\Delta t, h; p)$$
$$+ C\nu(\Delta t)^4 \left|\left|\left|\nabla u_{tt}\right|\right|\right|_{2,0}^2 + C\nu h^{2k} \left|\left|\left|u\right|\right|\right|_{2,k+1},$$

where

$$\begin{aligned} F(\Delta t, h; p) =& C\nu^{-1}h^{2k+1}\left(|||u|||_{4,k+1}^4 + |||\nabla u|||_{4,0}^4\right) \\ & + C\nu^{-1}h^{2k}\left(|||u|||_{4,k+1}^4 + \nu^{-1}\left(||u_h||^2 + \nu^{-1}|||f|||_{2,X'}^2\right)\right) \\ & + C\nu^{-1}\left(h^{2s+2}\left|\left|\left|p_{\frac{1}{2}}\right|\right|\right|_{2,s+1}^2 + (\Delta t)^4|||p_{tt}|||_{2,0}^2\right) \\ & + Ch^{2k+2}|||u_t|||_{2,k+1}^2 + C\Delta t h^{2k+2}||u_{tt}||_{L^2(0,T;H^{k+1})} \\ & + C(\Delta t)^4(|||u_{ttt}|||_{2,0}^2 + \nu^{-1}|||p_{tt}|||_{2,0}^2 + |||f_{tt}|||_{2,0}^2 + \nu|||\nabla u_{tt}|||_{2,0}^2 \\ & + \nu^{-1}|||\nabla u_{tt}|||_{4,0}^4 + \nu^{-1}|||\nabla u|||_{4,0}^4 + \nu^{-1}\left|\left|\left|\nabla u_{1/2}\right|\right|\right|_{4,0}^4) \end{aligned}$$

(ii) Then for all $0 \leq n \leq M$*, the following holds for EMAC,*

$$\begin{aligned} \left|\left|e^M\right|\right|^2 + \nu\Delta t\sum_{n=0}^{M-1}\left|\left|\nabla e^{n+\frac{1}{2}}\right|\right|^2 \leq& \exp\left(C\Delta t\sum_{n=0}^{M-1}\left(\nu^{-1}\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^3}^2 + \frac{3}{2}\right)\right)F(\Delta t, h; P) \\ & + C\nu(\Delta t)^4|||\nabla u_{tt}|||_{2,0}^2 + C\nu h^{2k}|||u|||_{2,2}, \end{aligned}$$

Proof. In order to properly give an error analysis, we need to match terms, so we will add the following terms to both sides of (2.16):

$$\left(\frac{u^{n+1} - u^n}{\Delta t}, v_h\right),\ b^*(u^{n+\frac{1}{2}}, u^{n+\frac{1}{2}}, v_h),\ \nu\left(\nabla u^{n+\frac{1}{2}}, \nabla v_h\right),\ -\left(p^{n+\frac{1}{2}}, \nabla \cdot v_h\right).$$

We will also add and subtract $\left(f^{n+\frac{1}{2}}, v_h\right)$ to the right hand side.

Applying this to (2.10) gives us

$$\begin{aligned} & \frac{1}{\Delta t}\left(u^{n+1} - u^n, v_h\right) + b^*(u^{n+\frac{1}{2}}, u^{n+\frac{1}{2}}, v_h) + \nu\left(\nabla u^{n+\frac{1}{2}}, \nabla v_h\right) - \left(p^{n+\frac{1}{2}}, \nabla \cdot v_h\right) \\ & = \left(f^{n+\frac{1}{2}}, v_h\right) + \tau(u^n, p^n; v_h), \end{aligned} \tag{3.1}$$

where

$$\tau(u^n,p^n;v_h) = \left(\frac{u^{n+1}-u^n}{\Delta t} - u_t(t^{n+\frac{1}{2}}), v_h\right) + \nu\left(\nabla u^{n+\frac{1}{2}} - \nabla u(t^{n+\frac{1}{2}}), \nabla v_h\right)$$
$$+ b^*(u^{n+\frac{1}{2}}, u^{n+\frac{1}{2}}, v_h) - b^*(u(t^{n+\frac{1}{2}}), u(t^{n+\frac{1}{2}}), v_h)$$
$$- \left(p^{n+\frac{1}{2}} - p(t^{n+\frac{1}{2}}), \nabla\cdot v_h\right) + \left(f(t^{n+\frac{1}{2}}) - f^{n+\frac{1}{2}}, v_h\right).$$

Subtract (2.16) from (3.1) to get

$$\begin{aligned}&\frac{1}{\Delta t}\left(e^{n+1}-e^n, v_h\right) + b^*(u^{n+\frac{1}{2}}, u^{n+\frac{1}{2}}, v_h) - b^*(u_h^{n+\frac{1}{2}}, u_h^{n+\frac{1}{2}}, v_h) + \nu\left(\nabla e^{n+\frac{1}{2}}, \nabla v_h\right)\\&- \left(p^{n+\frac{1}{2}} - q_h, \nabla\cdot v_h\right) = \tau(u^n,p^n;v_h).\end{aligned} \tag{3.2}$$

Note that $q_h \in V_h$, hence $(q_h, \nabla\cdot v_h) = 0$, which is why we may add that in (3.2). We now consider the Stokes operator (add reference and definition earlier) such that $\eta^n = u^n - I_h^{St}(u^n)$ and $\phi_h^n = I_h^{St}(u^n) - u_h^n$, which gives $e^n = \eta^n + \phi_h^n$. Apply this to (3.2), set $v_h = \phi_h^{n+\frac{1}{2}}$, and rearrange to get

$$\begin{aligned}&\left(\phi_h^{n+1} - \phi_h^n, \phi_h^{n+\frac{1}{2}}\right) + \nu\Delta t\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2\\=&-\Delta t\left(b^*(u^{n+\frac{1}{2}}, u^{n+\frac{1}{2}}, \phi_h^{n+\frac{1}{2}}) - b^*(u_h^{n+\frac{1}{2}}, u_h^{n+\frac{1}{2}}, \phi_h^{n+\frac{1}{2}})\right) - \left(\eta^{n+1} - \eta^n, \phi_h^{n+\frac{1}{2}}\right)\\&+ \Delta t\left(p^{n+\frac{1}{2}} - q_h, \nabla\cdot\phi_h^{n+\frac{1}{2}}\right) + \Delta t\tau(u^n,p^n;\phi_h^{n+\frac{1}{2}}).\end{aligned}$$

Note that $\left(\nabla\eta^{n+1}, \nabla\phi_h^{n+1}\right) = 0$ by us choosing $I_h^{St}(u)$ as our function in V_h to add and subtract in e^{n+1}. We may expand the first term on the left hand side, giving

$$\left(\phi_h^{n+1} - \phi_h^n, \phi_h^{n+\frac{1}{2}}\right) = \frac{1}{2}\left(\left|\left|\phi_h^{n+1}\right|\right|^2 - \left|\left|\phi_h^n\right|\right|^2\right).$$

Applying this gives us the equation

$$\begin{aligned}&\frac{1}{2}\left(\left|\left|\phi_h^{n+1}\right|\right|^2 - \left|\left|\phi_h^n\right|\right|^2\right) + \nu\Delta t\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2\\=&-\Delta t\left(b^*(u^{n+\frac{1}{2}}, u^{n+\frac{1}{2}}, \phi_h^{n+\frac{1}{2}}) - b^*(u_h^{n+\frac{1}{2}}, u_h^{n+\frac{1}{2}}, \phi_h^{n+\frac{1}{2}})\right) - \left(\eta^{n+1} - \eta^n, \phi_h^{n+\frac{1}{2}}\right)\\&+ \Delta t\left(p^{n+\frac{1}{2}} - q_h, \nabla\cdot\phi_h^{n+\frac{1}{2}}\right) + \Delta t\tau(u^n,p^n;\phi_h^{n+\frac{1}{2}}).\end{aligned} \tag{3.3}$$

Step 1: Bounds for terms in (3.3).

We first try and find bounds for the nonlinear terms in (3.3). We first rearrange them to get

$$\begin{aligned}
&b^*(u^{n+\frac{1}{2}},u^{n+\frac{1}{2}},\phi_h^{n+\frac{1}{2}})-b^*(u_h^{n+\frac{1}{2}},u_h^{n+\frac{1}{2}},\phi_h^{n+\frac{1}{2}})\\
=&b^*(\eta^{n+\frac{1}{2}},u^{n+\frac{1}{2}},\phi_h^{n+\frac{1}{2}})+b^*(\phi_h^{n+\frac{1}{2}},u^{n+\frac{1}{2}},\phi_h^{n+\frac{1}{2}})+b^*(u_h^{n+\frac{1}{2}},\eta^{n+\frac{1}{2}},\phi_h^{n+\frac{1}{2}}).
\end{aligned}$$

We now work to bound each of these terms on the right hand side. For each term, we use a standard trilinear term inequality and Young's inequality,

$$\begin{aligned}
b^*(\eta^{n+\frac{1}{2}},u^{n+\frac{1}{2}},\phi_h^{n+\frac{1}{2}})&\leq C\left|\left|\eta^{n+\frac{1}{2}}\right|\right|^{\frac{1}{2}}\left|\left|\nabla\eta^{n+\frac{1}{2}}\right|\right|^{\frac{1}{2}}\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|\\
&\leq\frac{\nu}{14}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2+C\nu^{-1}\left|\left|\eta^{n+\frac{1}{2}}\right|\right|\left|\left|\nabla\eta^{n+\frac{1}{2}}\right|\right|\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|^2.
\end{aligned}$$

For the second term we have,

$$\begin{aligned}
b^*(\phi_h^{n+\frac{1}{2}},u^{n+\frac{1}{2}},\phi_h^{n+\frac{1}{2}})&\leq C\left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^{\frac{1}{2}}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^{\frac{1}{2}}\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|\\
&\leq\frac{\nu}{14}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2+C\nu^{-3}\left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|^4.
\end{aligned}$$

Finally for the third term, we get

$$\begin{aligned}
b^*(u_h^{n+\frac{1}{2}},\eta^{n+\frac{1}{2}},\phi_h^{n+\frac{1}{2}})&\leq C\left|\left|u_h^{n+\frac{1}{2}}\right|\right|^{\frac{1}{2}}\left|\left|\nabla u_h^{n+\frac{1}{2}}\right|\right|^{\frac{1}{2}}\left|\left|\nabla\eta^{n+\frac{1}{2}}\right|\right|\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|\\
&\leq\frac{\nu}{14}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2+C\nu^{-1}\left|\left|u_h^{n+\frac{1}{2}}\right|\right|\left|\left|\nabla u_h^{n+\frac{1}{2}}\right|\right|\left|\left|\nabla\eta^{n+\frac{1}{2}}\right|\right|^2.
\end{aligned}$$

Following eq (4.14) from [18], we get the following bound

$$\frac{1}{\Delta t}\left(\eta^{n+1}-\eta^n,\phi_h^{n+\frac{1}{2}}\right)\leq\left|\left|\eta_t^{n+1}\right|\right|^2+\int_{t^n}^{t^{n+1}}\left|\left|\eta_{tt}\right|\right|^2dt+\frac{1}{2}\left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2.$$

Then for the pressure term we use the Cauchy Schwarz and Young's inequality,

$$\begin{aligned}
\left|\left(p^{n+\frac{1}{2}}-q_h,\nabla\cdot\phi_h^{n+\frac{1}{2}}\right)\right|&\leq C\left|\left|p^{n+\frac{1}{2}}-q_h\right|\right|\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|\\
&\leq\frac{\nu}{14}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2+C\nu^{-1}\left|\left|p^{n+\frac{1}{2}}-q_h\right|\right|^2.
\end{aligned}$$

Now we combine all of these bounds into (3.3) and get

$$\begin{aligned}
&\frac{1}{2}\left|\left|\phi_h^M\right|\right|^2 + \nu\Delta t\sum_{n=0}^{M-1}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right| \\
\leq & C\nu^{-3}\Delta t\sum_{n=0}^{M-1}\left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|^4 + \frac{4\nu}{14}\Delta t\sum_{n=0}^{M-1}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2 \\
&+ C\nu^{-1}\Delta t\sum_{n=0}^{M-1}\left|\left|\eta^{n+\frac{1}{2}}\right|\right|\left|\left|\nabla\eta^{n+\frac{1}{2}}\right|\right|\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|^2 \\
&+ C\nu^{-1}\Delta t\sum_{n=0}^{M-1}\left|\left|u_h^{n+\frac{1}{2}}\right|\right|\left|\left|\nabla u_h^{n+\frac{1}{2}}\right|\right|\left|\left|\nabla\eta^{n+\frac{1}{2}}\right|\right|^2 \\
&+ \Delta t\sum_{n=0}^{M-1}\left|\left|\eta_t^{n+1}\right|\right|^2 + \left|\left|\eta_{tt}\right|\right|^2_{L^2(0,T;L^2)} + \frac{\Delta t}{2}\sum_{n=0}^{M-1}\left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2 \\
&+ C\nu^{-1}\Delta t\sum_{n=0}^{M-1}\left|\left|p^{n+\frac{1}{2}} - q_h\right|\right|^2 + \Delta t\sum_{n=0}^{M-1}C|\tau(u^n, p^n; \phi_h^{n+\frac{1}{2}})|.
\end{aligned} \tag{3.4}$$

Step 2: Bounds for terms in (3.4).

The second and third term are bounded by expanding out the midpoint terms and applying Young's inequality, specific details can be found in [61].

$$\begin{aligned}
&C\nu^{-1}\Delta t\sum_{n=0}^{M-1}\left|\left|\eta^{n+\frac{1}{2}}\right|\right|\left|\left|\nabla\eta^{n+\frac{1}{2}}\right|\right|\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|^2 \leq C\nu^{-1}h^{2k+1}\left(|||u|||^4_{4,k+1} + |||\nabla u|||^4_{4,0}\right), \\
&C\nu^{-1}\Delta t\sum_{n=0}^{M-1}\left|\left|u_h^{n+\frac{1}{2}}\right|\right|\left|\left|\nabla u_h^{n+\frac{1}{2}}\right|\right|\left|\left|\nabla\eta^{n+\frac{1}{2}}\right|\right|^2 \\
&\quad\leq C\nu^{-1}h^{2k}\left(|||u|||^4_{4,k+1} + \nu^{-1}\left(||u_h||^2 + \nu^{-1}|||f|||^2_{2,X'}\right)\right).
\end{aligned}$$

For the pressure term, we get

$$C\nu^{-1}\Delta t\sum_{n=0}^{M-1}\left|\left|p^{n+\frac{1}{2}} - q_h\right|\right|^2 \leq C\nu^{-1}\left(h^{2s+2}\left|\left|\left|p_{\frac{1}{2}}\right|\right|\right|^2_{2,s+1} + (\Delta t)^4|||p_{tt}|||^2_{2,0}\right).$$

For the time derivative term, we apply the approximation result (2.3)

$$\Delta t\sum_{n=0}^{M-1}\left|\left|\eta_t^{n+1}\right|\right|^2 \leq Ch^{2k+2}|||u_t|||^2_{2,k+1}.$$

For the final term, we get

$$\Delta t \left|\left|\eta_{tt}\right|\right|^2_{L^2(0,T;L^2)} \leq C\Delta t h^{2k+2} \left|\left|u_{tt}\right|\right|^2_{L^2(0,T;H^{k+1})}.$$

Combining all of the previous bounds gives us

$$\begin{aligned}
&\left|\left|\phi_h^M\right|\right|^2 + \nu\Delta t \sum_{n=0}^{M-1} \left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2 \\
\leq & C\Delta t \sum_{n=0}^{M-1} \left(\nu^{-3} \left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|^4 + \frac{1}{2}\right) \left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2 + \frac{4\nu}{14}\Delta t \sum_{n=0}^{M-1} \left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2 \\
&+ C\nu^{-1}h^{2k+1}\left(|||u|||^4_{4,k+1} + |||\nabla u|||^4_{4,0}\right) \\
&+ C\nu^{-1}h^{2k}\left(|||u|||^4_{4,k+1} + \nu^{-1}\left(||u_h||^2 + \nu^{-1}\,|||f|||^2_{2,X'}\right)\right) \\
&+ C\nu^{-1}\left(h^{2s+2}\left|\left|p_{\frac{1}{2}}\right|\right|^2_{2,s+1} + (\Delta t)^4\,|||p_{tt}|||^2_{2,0}\right) + Ch^{2k+2}\,|||u_t|||^2_{2,k+1} \\
&+ C\Delta t h^{2k+2}\left|\left|u_{tt}\right|\right|_{L^2(0,T;H^{k+1})} + \Delta t \sum_{n=0}^{M-1} C|\tau(u^n,p^n;\phi_h^{n+\frac{1}{2}})|.
\end{aligned} \tag{3.5}$$

Step 3: Bounds for τ.

Now to handle each of the terms in $\tau(u^n,p^n;\phi_h^{n+\frac{1}{2}})$. For the first term, we use Cauchy-Schwarz and Young's inequality along with a known Taylor series result

$$\begin{aligned}
\left(\frac{u^{n+1}-u^n}{\Delta t} - u_t(t^{n+\frac{1}{2}}), \phi_h^{n+\frac{1}{2}}\right) &\leq \frac{1}{2}\left|\left|\phi_h^{n+\frac{1}{2}}\right|\right| + \frac{1}{2}\left|\left|\frac{u^{n+1}-u^n}{\Delta t} - u_t(t^{n+\frac{1}{2}})\right|\right|^2 \\
&\leq \frac{1}{2}\left|\left|\phi_h^{n+1}\right|\right|^2 + \frac{1}{2}\left|\left|\phi_h^n\right|\right|^2 + \frac{1}{2}\frac{(\Delta t)^3}{1280}\int_{t^n}^{t^{n+1}} \left|\left|u_{ttt}\right|\right|^2 dt.
\end{aligned}$$

For the pressure term, we get

$$\begin{aligned}
\left(p^{n+\frac{1}{2}} - p(t^{n+\frac{1}{2}}), \nabla\cdot\phi_h^{n+\frac{1}{2}}\right) &\leq \frac{\nu}{14}\left|\left|\nabla\phi^{n+\frac{1}{2}}\right|\right|^2 + C\nu^{-1}\left|\left|p^{n+\frac{1}{2}} - p(t^{n+\frac{1}{2}})\right|\right|^2 \\
&\leq \frac{\nu}{14}\left|\left|\nabla\phi^{n+\frac{1}{2}}\right|\right|^2 + C\nu^{-1}\frac{(\Delta t)^3}{48}\int_{t^n}^{t^{n+1}} \left|\left|p_{tt}\right|\right|^2 dt.
\end{aligned}$$

For the forcing function, we have

$$\begin{aligned}\left(f(t^{n+\frac{1}{2}})-f^{n+\frac{1}{2}},\phi_h^{n+\frac{1}{2}}\right) &\leq \frac{1}{2}\left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2+\frac{1}{2}\left|\left|f(t^{n+\frac{1}{2}})-f^{n+\frac{1}{2}}\right|\right|^2\\ &\leq \frac{1}{2}\left|\left|\phi_h^{n+1}\right|\right|^2+\frac{1}{2}\left|\left|\phi_h^{n}\right|\right|^2+\frac{(\Delta t)^3}{48}\int_{t^n}^{t^{n+1}}\left|\left|f_{tt}\right|\right|^2 dt.\end{aligned}$$

Through similar technique, we get

$$\begin{aligned}\left(\nabla u^{n+\frac{1}{2}}-\nabla u(t^{n+\frac{1}{2}}),\nabla\phi_h^{n+\frac{1}{2}}\right) &\leq \frac{\nu}{14}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2+C\nu\left|\left|\nabla u^{n+\frac{1}{2}}-\nabla u(t^{n+\frac{1}{2}})\right|\right|^2\\ &\leq \frac{\nu}{14}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2+C\nu\frac{(\Delta t)^3}{48}\int_{t^n}^{t^{n+1}}\left|\left|\nabla u_{tt}\right|\right|^2 dt.\end{aligned}$$

Now to find a bound for the nonlinear terms,

$$\begin{aligned}&b^*(u^{n+\frac{1}{2}},u^{n+\frac{1}{2}},\phi_h^{n+\frac{1}{2}})-b^*(u(t^{n+\frac{1}{2}}),u(t^{n+\frac{1}{2}}),\phi_h^{n+\frac{1}{2}})\\ =&b^*(u^{n+\frac{1}{2}}-u(t^{n+\frac{1}{2}}),u^{n+\frac{1}{2}},\phi_h^{n+\frac{1}{2}})-b^*(u(t^{n+\frac{1}{2}}),u^{n+\frac{1}{2}}-u(t^{n+\frac{1}{2}}),\phi_h^{n+\frac{1}{2}})\\ \leq&C\left|\left|\nabla\left(u^{n+\frac{1}{2}}-u(t^{n+\frac{1}{2}})\right)\right|\right|\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|\left(\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|+\left|\left|\nabla u(t^{n+\frac{1}{2}})\right|\right|\right).\end{aligned}$$

Then apply Young's inequality and the estimate on the first quantity on the right hand side,

$$\begin{aligned}\leq&C\nu^{-1}\left(\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|^2+\left|\left|\nabla u(t^{n+\frac{1}{2}})\right|\right|^2\right)\frac{(\Delta t)^3}{48}\int_{t^n}^{t^{n+1}}\left|\left|\nabla u_{tt}\right|\right|^2 dt+\frac{\nu}{14}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2\\ \leq&C\nu^{-1}(\Delta t)^3\left(\int_{t^n}^{t^{n+1}}\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|^4+\left|\left|\nabla u(t^{n+\frac{1}{2}})\right|\right|^4+\left|\left|\nabla u_{tt}\right|\right|^4 dt\right)+\frac{\nu}{14}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2\\ \leq&C\nu^{-1}(\Delta t)^4\left(\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|^4+\left|\left|\nabla u(t^{n+\frac{1}{2}})\right|\right|^4\right)+C\nu^{-1}(\Delta t)^3\int_{t^n}^{t^{n+1}}\left|\left|\nabla u_{tt}\right|\right|^4 dt+\frac{\nu}{14}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2.\end{aligned}$$

Now we can combine the previous five estimates and add them together to get

$$\begin{aligned}\Delta t\sum_{n=0}^{M-1}\left|\tau(u^n,p^n;v_h)\right| \leq C\Delta t\sum_{n=0}^{M}\left|\left|\phi_h^n\right|\right|^2+\frac{3\nu}{14}\Delta t\sum_{n=0}^{M-1}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2&\\ +C(\Delta t)^4(|||u_{ttt}|||_{2,0}^2+\nu^{-1}|||p_{tt}|||_{2,0}^2+|||f_{tt}|||_{2,0}^2+\nu|||\nabla u_{tt}|||_{2,0}^2&\\ +\nu^{-1}|||\nabla u_{tt}|||_{4,0}^4+\nu^{-1}|||\nabla u|||_{4,0}^4+\nu^{-1}\left|\left|\left|\nabla u_{1/2}\right|\right|\right|_{4,0}^4).&\end{aligned} \tag{3.6}$$

Finally, use (3.6) in (3.5) and combine terms to get

$$\begin{aligned}
&||\phi_h^M||^2 + \nu\Delta t \sum_{n=0}^{M-1} \left|\left|\nabla \phi_h^{n+\frac{1}{2}}\right|\right|^2 \\
\leq &C\Delta t \sum_{n=0}^{M-1} \left(\nu^{-3} \left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|^4 + \frac{3}{2}\right) \left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2 \\
&+ C\nu^{-1} h^{2k+1} \left(|||u|||_{4,k+1}^4 + |||\nabla u|||_{4,0}^4 \right) \\
&+ C\nu^{-1} h^{2k} \left(|||u|||_{4,k+1}^4 + \nu^{-1} \left(||u_h||^2 + \nu^{-1} |||f|||_{2,X'}^2 \right) \right) \\
&+ C\nu^{-1} \left(h^{2s+2} \left|\left|\left|p_{\frac{1}{2}}\right|\right|\right|_{2,s+1}^2 + (\Delta t)^4 |||p_{tt}|||_{2,0}^2 \right) \\
&+ Ch^{2k+2} |||u_t|||_{2,k+1}^2 + C\Delta t h^{2k+2} ||u_{tt}||_{L^2(0,T;H^{k+1})} \\
&+ C(\Delta t)^4 (|||u_{ttt}|||_{2,0}^2 + \nu^{-1} |||p_{tt}|||_{2,0}^2 + |||f_{tt}|||_{2,0}^2 + \nu |||\nabla u_{tt}|||_{2,0}^2 \\
&+ \nu^{-1} |||\nabla u_{tt}|||_{4,0}^4 + \nu^{-1} |||\nabla u|||_{4,0}^4 + \nu^{-1} \left|\left|\left|\nabla u_{1/2}\right|\right|\right|_{4,0}^4).
\end{aligned} \tag{3.7}$$

Here we set

$$\begin{aligned}
F(\Delta t, h; p) = &C\nu^{-1} h^{2k+1} \left(|||u|||_{4,k+1}^4 + |||\nabla u|||_{4,0}^4 \right) \\
&+ C\nu^{-1} h^{2k} \left(|||u|||_{4,k+1}^4 + \nu^{-1} \left(||u_h||^2 + \nu^{-1} |||f|||_{2,X'}^2 \right) \right) \\
&+ C\nu^{-1} \left(h^{2s+2} \left|\left|\left|p_{\frac{1}{2}}\right|\right|\right|_{2,s+1}^2 + (\Delta t)^4 |||p_{tt}|||_{2,0}^2 \right) \\
&+ Ch^{2k+2} |||u_t|||_{2,k+1}^2 + C\Delta t h^{2k+2} ||u_{tt}||_{L^2(0,T;H^{k+1})} \\
&+ C(\Delta t)^4 (|||u_{ttt}|||_{2,0}^2 + \nu^{-1} |||p_{tt}|||_{2,0}^2 + |||f_{tt}|||_{2,0}^2 + \nu |||\nabla u_{tt}|||_{2,0}^2 \\
&+ \nu^{-1} |||\nabla u_{tt}|||_{4,0}^4 + \nu^{-1} |||\nabla u|||_{4,0}^4 + \nu^{-1} \left|\left|\left|\nabla u_{1/2}\right|\right|\right|_{4,0}^4)
\end{aligned}$$

and apply Lemma 6 to (3.7) to get

$$||\phi_h^M||^2 + \nu\Delta t \sum_{n=0}^{M-1} \left|\left|\nabla \phi_h^{n+\frac{1}{2}}\right|\right|^2 \leq \exp\left(C\Delta t \sum_{n=0}^{M-1} \left(\nu^{-3} \left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|^4 + \frac{3}{2} \right) \right) F(\Delta t, h; p).$$

Using the triangle inequality from here gives us the desired result.

Proving part (ii) of this theorem is nearly identical to the proof for part (i). The only differences are the nonlinear terms, however the nonlinear terms in $\tau(u^n, p^n; \phi_h^{n+\frac{1}{2}})$ can be bounded identically to how it was bounded for SKEW. So the only thing that needs bounded is the term

$$c(u^{n+\frac{1}{2}}, u^{n+\frac{1}{2}}, \phi_h^{n+\frac{1}{2}}) - c(u_h^{n+\frac{1}{2}}, u_h^{n+\frac{1}{2}}, \phi_h^{n+\frac{1}{2}}).$$

The bound is found by manipulating the symmetric difference and using the Stokes operator, which can be found in detail in [68]. The bound is as follows,

$$\begin{aligned}
&|c(u^{n+\frac{1}{2}}, u^{n+\frac{1}{2}}, \phi_h^{n+\frac{1}{2}}) - c(u_h^{n+\frac{1}{2}}, u_h^{n+\frac{1}{2}}, \phi_h^{n+\frac{1}{2}})| \\
\leq & C\nu^{-1}\left(\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|^2 \left|\left|\eta^{n+\frac{1}{2}}\right|\right| \left|\left|\nabla \eta^{n+\frac{1}{2}}\right|\right| + \left|\left|\nabla u^{n+\frac{1}{2}}\right|\right| \left|\left|u^{n+\frac{1}{2}}\right|\right| \left|\left|\nabla \eta^{n+\frac{1}{2}}\right|\right|^2 \right) \\
&+ C\nu^{-1}\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^3}^2 \left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2 + \frac{3\nu}{14}\left|\left|\nabla \phi_h^{n+\frac{1}{2}}\right|\right|^2.
\end{aligned}$$

Then we get

$$\begin{aligned}
&\left|\left|\phi_h^M\right|\right|^2 + \nu\Delta t \sum_{n=0}^{M-1}\left|\left|\nabla \phi_h^{n+\frac{1}{2}}\right|\right|^2 \\
\leq & C\Delta t \sum_{n=0}^{M-1}\left(\nu^{-1}\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^3}^2 + \frac{3}{2}\right)\left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2 + F(\Delta t, h; P).
\end{aligned}$$

Applying the Gronwall inequality, we get

$$\left|\left|\phi_h^M\right|\right|^2 + \nu\Delta t \sum_{n=0}^{M-1}\left|\left|\nabla \phi_h^{n+\frac{1}{2}}\right|\right|^2 \leq \exp\left(C\Delta t \sum_{n=0}^{M-1}\left(\nu^{-1}\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^3}^2 + \frac{3}{2}\right)\right) F(\Delta t, h; P).$$

The triangle inequality gives us the final result of the proof. ■

3.2 Analysis with $H^3(\Omega)$ assumed velocity regularity

Now we give the fully discrete analysis with slightly stronger requirements on the velocity terms.

Theorem 8. *Let* (u_h^{n+1}, p_h^{n+1}) *solve* (2.16)-(2.17) *and* (u^{n+1}, p^{n+1}) *solve* (2.10)-(2.11) *be a NSE solution with* $u_t^{n+1} \in X'$, $u^{n+1} \in H^3(\Omega)$, *and* $p^{n+1} \in H^2(\Omega)$, *for* $0 \leq n \leq M$. *Additionally, we assume* $u^{n+\frac{1}{2}}, \nabla u^{n+\frac{1}{2}} \in L^\infty(\Omega)$. *Denote* $e^n = u^n - u_h^n$, $\eta^n = u^n - I_h^{St}(u^n)$, *and* $\phi_h^n = I_h^{St}(u^n) - u_h^n$.

(i) Then for all $0 \leq n \leq M$*, the following holds for SKEW*

$$\begin{aligned}
&\left|\left|e^M\right|\right|^2 + \nu\Delta t \sum_{n=0}^{M-1} \left|\left|\nabla e^{n+\frac{1}{2}}\right|\right|^2 \\
\leq &\exp\left(C\Delta t \sum_{n=0}^{M-1} \left(\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^\infty} + \nu^{-1} \left|\left|u^{n+\frac{1}{2}}\right|\right|^2_{L^\infty} + \frac{3}{2} \right)\right) F(\Delta t, h) \\
&+ C\nu(\Delta t)^4 \left|\left|\left|\nabla u_{tt}\right|\right|\right|^2_{2,0} + C\nu h^{2k} \left|\left|\left|u\right|\right|\right|_{2,k+1},
\end{aligned}$$

where

$$\begin{aligned}
F(\Delta t, h) =&C\nu^{-1} h^{2k+1} \left(|||u|||^4_{4,k+1} + |||\nabla u|||^4_{4,0} \right) \\
&+ C\nu^{-1} h^{2k} \left(|||u|||^4_{4,k+1} + \nu^{-1} \left(||u_h||^2 + \nu^{-1} |||f|||^2_{2,X'} \right) \right) \\
&+ C\nu^{-1} \left(h^{2s+2} \left|\left|\left|p_{\frac{1}{2}}\right|\right|\right|^2_{2,s+1} + (\Delta t)^4 |||p_{tt}|||^2_{2,0} \right) \\
&+ Ch^{2k+2} |||u_t|||^2_{2,k+1} + C\Delta t h^{2k+2} ||u_{tt}||^2_{L^2(0,T;H^{k+1})} \\
&+ C(\Delta t)^4 (|||u_{ttt}|||^2_{2,0} + \nu^{-1} |||p_{tt}|||^2_{2,0} + |||f_{tt}|||^2_{2,0} + \nu |||\nabla u_{tt}|||^2_{2,0} \\
&+ \nu^{-1} |||\nabla u_{tt}|||^4_{4,0} + \nu^{-1} |||\nabla u|||^4_{4,0} + \nu^{-1} \left|\left|\left|\nabla u_{1/2}\right|\right|\right|^4_{4,0}).
\end{aligned}$$

(ii) Then for all $0 \leq n \leq M$*, the following holds for EMAC,*

$$\begin{aligned}
&\left|\left|e^M\right|\right|^2 + \nu\Delta t \sum_{n=0}^{M-1} \left|\left|\nabla e^{n+\frac{1}{2}}\right|\right|^2 \\
\leq &\exp\left(C\Delta t \sum_{n=0}^{M-1} \left(\left|\left|u^{n+\frac{1}{2}}\right|\right|_{L^\infty} + \left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^\infty} + \frac{3}{2} \right)\right) G(\Delta t, h; P) \\
&+ C\nu(\Delta t)^4 |||\nabla u_{tt}|||^2_{2,0} + C\nu h^{2k} |||u|||^2_{2,k+1},
\end{aligned}$$

where

$$\begin{aligned}
G(\Delta t, h) = {} & Ch^{2k}\left(|||u|||_{2,\infty}^2 + |||u|||_{4,k+1}^4\right) \\
& + Ch^{2k+1}\left(|||\nabla u|||_{2,\infty}^2 + |||u|||_{4,k+1}^4\right) \\
& + C\nu^{-1}\left(h^{2s+2}\left|\left|\left|P_{\frac{1}{2}}\right|\right|\right|_{2,s+1}^2 + (\Delta t)^4\,|||P_{tt}|||_{2,0}^2\right) \\
& + Ch^{2k+2}\,|||u_t|||_{2,k+1}^2 + C\Delta t h^{2k+2}\,||u_{tt}||_{L^2(0,T;H^{k+1})}^2 \\
& + C(\Delta t)^4(|||u_{ttt}|||_{2,0}^2 + \nu^{-1}\,|||P_{tt}|||_{2,0}^2 + |||f_{tt}|||_{2,0}^2 + \nu\,|||\nabla u_{tt}|||_{2,0}^2 \\
& + \nu^{-1}\,|||\nabla u_{tt}|||_{4,0}^4 + \nu^{-1}\,|||\nabla u|||_{4,0}^4 + \nu^{-1}\,|||\nabla u_{1/2}|||_{4,0}^4).
\end{aligned}$$

Proof. The proof for this theorem follows similarly to Theorem 7. The key difference is how we handle the term $b^*(\phi_h^{n+\frac{1}{2}}, u^{n+\frac{1}{2}}, \phi_h^{n+\frac{1}{2}})$. We apply Hölder's and Young's inequality to get

$$\begin{aligned}
|b^*(\phi_h^{n+\frac{1}{2}}, u^{n+\frac{1}{2}}, \phi_h^{n+\frac{1}{2}})| &= \left|b(\phi_h^{n+\frac{1}{2}}, u^{n+\frac{1}{2}}, \phi_h^{n+\frac{1}{2}}) + \frac{1}{2}\left((\nabla\cdot\phi_h^{n+\frac{1}{2}})\phi_h^{n+\frac{1}{2}}, u^{n+\frac{1}{2}}\right)\right| \\
&\leq \left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^\infty}\left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2 + \frac{1}{2}\left|\left|u^{n+\frac{1}{2}}\right|\right|_{L^\infty}\left|\left|\nabla\cdot\phi_h^{n+\frac{1}{2}}\right|\right|\left|\left|\phi_h^{n+\frac{1}{2}}\right|\right| \\
&\leq \left(\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^\infty} + \nu^{-1}\left|\left|u^{n+\frac{1}{2}}\right|\right|_{L^\infty}^2\right)\left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2 + \frac{\nu}{14}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2.
\end{aligned}$$

The rest of the proof follows identically to the proof for Theorem 7 with the exception of the Gronwall constant term, so we end with

$$\begin{aligned}
& \left|\left|\phi_h^M\right|\right|^2 + \nu\Delta t\sum_{n=0}^{M-1}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2 \\
\leq{} & C\Delta t\sum_{n=0}^{M-1}\left(\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^\infty} + \nu^{-1}\left|\left|u^{n+\frac{1}{2}}\right|\right|_{L^\infty}^2 + \frac{3}{2}\right)\left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2 \\
& + C\nu^{-1}h^{2k+1}\left(|||u|||_{4,k+1}^4 + |||\nabla u|||_{4,0}^4\right) \\
& + C\nu^{-1}h^{2k}\left(|||u|||_{4,k+1}^4 + \nu^{-1}\left(||u_h||^2 + \nu^{-1}\,|||f|||_{2,X'}^2\right)\right) \\
& + C\nu^{-1}\left(h^{2s+2}\left|\left|\left|p_{\frac{1}{2}}\right|\right|\right|_{2,s+1}^2 + (\Delta t)^4\,|||p_{tt}|||_{2,0}^2\right) \\
& + Ch^{2k+2}\,|||u_t|||_{2,k+1}^2 + C\Delta t h^{2k+2}\,||u_{tt}||_{L^2(0,T;H^{k+1})} \\
& + C(\Delta t)^4(|||u_{ttt}|||_{2,0}^2 + \nu^{-1}\,|||p_{tt}|||_{2,0}^2 + |||f_{tt}|||_{2,0}^2 + \nu\,|||\nabla u_{tt}|||_{2,0}^2 \\
& + \nu^{-1}\,|||\nabla u_{tt}|||_{4,0}^4 + \nu^{-1}\,|||\nabla u|||_{4,0}^4 + \nu^{-1}\,|||\nabla u_{1/2}|||_{4,0}^4).
\end{aligned} \tag{3.8}$$

Here we set

$$\begin{aligned}F(\Delta t, h) =& C\nu^{-1}h^{2k+1}\left(|||u|||_{4,k+1}^4 + |||\nabla u|||_{4,0}^4\right)\\ &+ C\nu^{-1}h^{2k}\left(|||u|||_{4,k+1}^4 + \nu^{-1}\left(||u_h||^2 + \nu^{-1}|||f|||_{2,X'}^2\right)\right)\\ &+ C\nu^{-1}\left(h^{2s+2}\left|\left|\left|p_{\frac{1}{2}}\right|\right|\right|_{2,s+1}^2 + (\Delta t)^4|||p_{tt}|||_{2,0}^2\right)\\ &+ Ch^{2k+2}|||u_t|||_{2,k+1}^2 + C\Delta t h^{2k+2}||u_{tt}||_{L^2(0,T;H^{k+1})}\\ &+ C(\Delta t)^4(|||u_{ttt}|||_{2,0}^2 + \nu^{-1}|||p_{tt}|||_{2,0}^2 + |||f_{tt}|||_{2,0}^2 + \nu|||\nabla u_{tt}|||_{2,0}^2\\ &+ \nu^{-1}|||\nabla u_{tt}|||_{4,0}^4 + \nu^{-1}|||\nabla u|||_{4,0}^4 + \nu^{-1}\left|\left|\left|\nabla u_{1/2}\right|\right|\right|_{4,0}^4).\end{aligned}$$

Apply Lemma 6 to (3.8) and get

$$\begin{aligned}&\left|\left|\phi_h^M\right|\right|^2 + \nu\Delta t\sum_{n=0}^{M-1}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2\\ \leq&\exp\left(C\Delta t\sum_{n=0}^{M-1}\left(\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^\infty} + \nu^{-1}\left|\left|u^{n+\frac{1}{2}}\right|\right|_{L^\infty}^2 + \frac{3}{2}\right)\right)F(\Delta t, h).\end{aligned}$$

Using the triangle inequality from here gives us the desired result.

Proving part (ii) of this theorem is nearly identical to the proof for part (ii) for Theorem 7. We bound the nonlinear term by manipulating the symmetric difference and using Stokes operator, which can be found in detail in [68]. The bound is as follows,

$$\begin{aligned}&|c(u^{n+\frac{1}{2}}, u^{n+\frac{1}{2}}, \phi_h^{n+\frac{1}{2}}) - c(u_h^{n+\frac{1}{2}}, u_h^{n+\frac{1}{2}}, \phi_h^{n+\frac{1}{2}})|\\ \leq& C\left(\left|\left|\nabla\eta^{n+\frac{1}{2}}\right|\right|^2\left|\left|u^{n+\frac{1}{2}}\right|\right|_{L^\infty} + \left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^\infty}\left|\left|\eta^{n+\frac{1}{2}}\right|\right|^2\right.\\ &\left.+\left(\left|\left|u^{n+\frac{1}{2}}\right|\right|_{L^\infty} + \left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^\infty}\right)\left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2\right).\end{aligned}$$

We then get the following bound

$$\begin{aligned}
&\frac{1}{2}\left|\left|\phi_h^M\right|\right|^2+\nu\Delta t\sum_{n=0}^{M-1}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right| \\
\leq & C\nu^{-3}\Delta t\sum_{n=0}^{M-1}\left(\left|\left|u^{n+\frac{1}{2}}\right|\right|_{L^\infty}+\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^\infty}\right)\left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2 \\
&+\frac{4\nu}{14}\Delta t\sum_{n=0}^{M-1}\left|\left|\nabla\phi_h^{n+\frac{1}{2}}\right|\right|^2 \\
&+C\Delta t\sum_{n=0}^{M-1}\left|\left|\nabla\eta^{n+\frac{1}{2}}\right|\right|^2\left|\left|u^{n+\frac{1}{2}}\right|\right|_{L^\infty} \\
&+C\Delta t\sum_{n=0}^{M-1}\left|\left|\eta^{n+\frac{1}{2}}\right|\right|^2\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^\infty} \\
&+\Delta t\sum_{n=0}^{M-1}\left|\left|\eta_t^{n+1}\right|\right|^2+\left|\left|\eta_{tt}\right|\right|_{L^2(0,T;L^2)}^2+\frac{\Delta t}{2}\sum_{n=0}^{M-1}\left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2 \\
&+C\nu^{-1}\Delta t\sum_{n=0}^{M-1}\left|\left|P^{n+\frac{1}{2}}-q_h\right|\right|^2+\Delta t\sum_{n=0}^{M-1}C|\tau(u^n,p^n;\phi_h^{n+\frac{1}{2}})|.
\end{aligned}$$

We have proper bounds for each of the terms except for the two from the EMAC trilinear terms, which are easily bounded using similar techniques that we've seen earlier in the proof

$$\begin{aligned}
C\Delta t\sum_{n=0}^{M-1}\left|\left|\nabla\eta^{n+\frac{1}{2}}\right|\right|^2\left|\left|u^{n+\frac{1}{2}}\right|\right|_{L^\infty} &\leq Ch^{2k}\left(|||u|||_{2,\infty}^2+|||u|||_{4,k+1}^4\right), \\
C\Delta t\sum_{n=0}^{M-1}\left|\left|\eta^{n+\frac{1}{2}}\right|\right|^2\left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^\infty} &\leq Ch^{2k+1}\left(|||\nabla u|||_{2,\infty}^2+|||u|||_{4,k+1}^4\right).
\end{aligned}$$

Combining this with all of the terms gives

$$\begin{aligned}
&\left|\left|\phi_h^M\right|\right|^2 + \nu\Delta t \sum_{n=0}^{M-1} \left|\left|\nabla \phi_h^{n+\frac{1}{2}}\right|\right|^2 \\
\leq & C\Delta t \sum_{n=0}^{M-1} \left(\left|\left|u^{n+\frac{1}{2}}\right|\right|_{L^\infty} + \left|\left|\nabla u^{n+\frac{1}{2}}\right|\right|_{L^\infty} + \frac{3}{2}\right) \left|\left|\phi_h^{n+\frac{1}{2}}\right|\right|^2 \\
&+ Ch^{2k} \left(|||u|||_{2,\infty}^2 + |||u|||_{4,k+1}^4 \right) \\
&+ Ch^{2k+1} \left(|||\nabla u|||_{2,\infty}^2 + |||u|||_{4,k+1}^4 \right) \\
&+ C\nu^{-1} \left(h^{2s+2} \left|\left|\left|P_{\frac{1}{2}}\right|\right|\right|_{2,s+1}^2 + (\Delta t)^4 \, |||P_{tt}|||_{2,0}^2 \right) \\
&+ Ch^{2k+2} \, |||u_t|||_{2,k+1}^2 + C\Delta t h^{2k+2} \, ||u_{tt}||_{L^2(0,T;H^{k+1})} \\
&+ C(\Delta t)^4 (|||u_{ttt}|||_{2,0}^2 + \nu^{-1} \, |||P_{tt}|||_{2,0}^2 + |||f_{tt}|||_{2,0}^2 + \nu \, |||\nabla u_{tt}|||_{2,0}^2 \\
&+ \nu^{-1} \, |||\nabla u_{tt}|||_{4,0}^4 + \nu^{-1} \, |||\nabla u|||_{4,0}^4 + \nu^{-1} \left|\left|\left|\nabla u_{1/2}\right|\right|\right|_{4,0}^4).
\end{aligned}$$

The result easily follows. ■

3.3 Numerical tests

In this section, we provide numerical results that reinforce the strengths of EMAC over SKEW, in particular the longer time accuracy of EMAC suggested by the better Gronwall constant in its convergence analysis. We test both coupled schemes for SKEW and EMAC, as well as projection methods. We use Freefem++ [44] to perform these simulation. Newton iterations are used to resolve the nonlinearities. Here we will define the projection methods used for comparison [52] that will be included in our numerical tests. We begin by defining the spaces

$$Y = \left\{ v \in L^2(\Omega),\ \nabla \cdot v \in L^2(\Omega),\ v \cdot \hat{n}|_{\partial\Omega} = 0 \right\}$$

and the discrete subspace $Y_h = Y \cap P_k(\tau_h)$. Our projection methods are as follows:

Step 1 EMAC-BE-PROJ: Find $\tilde{u}_h^{n+1} \in X_h$ satisfying, for all $\chi_h \in X_h$,

$$\frac{1}{\Delta t}(\tilde{u}_h^{n+1} - u_h^n, \chi_h) + c(\tilde{u}_h^{n+1}, \tilde{u}_h^{n+1}, \chi_h) + \nu(\nabla \tilde{u}_h^{n+1}, \nabla \chi_h) = (f(t^{n+1}), \chi_h). \tag{3.9}$$

Step 1 SKEW-BE-PROJ is the same as Step 1 EMAC-BE-PROJ except the nonlinear term.

Step 2: Find $(u_h^{n+1}, P_h^{n+1}) \in (Y_h, Q_h)$ satisfying, for all $(w_h, q_h) \in (Y_h, Q_h)$,

$$\begin{cases} \frac{1}{\Delta t}(u_h^{n+1}, w_h) - (P_h^{n+1}, \nabla \cdot w_h) & = \frac{1}{\Delta t}(\tilde{u}_h^{n+1}, w_h), \\ (\nabla \cdot {u_h}^{n+1}, q_h) & = 0. \end{cases} \tag{3.10}$$

More details on the analysis and derivation for these two schemes can be found in Chapters 3 and 4 in [52].

Step 1 RotProjB-EMAC: Find $\tilde{u}_h^{n+1} \in X_h$ satisfying, for all $\chi_h \in X_h$,

$$\begin{cases} \frac{1}{\Delta t}\left(\tilde{u}_h^{n+1} - u_h^n, \chi_h\right) + \frac{1}{4}c(\tilde{u}_h^{n+1} + u_h^n, \tilde{u}^{n+1} + u_h^n, \chi_h) + \frac{\nu}{2}\left(\nabla(\tilde{u}_h^{n+1} + u^n), \nabla \chi_h\right) \\ \quad -(p_h^n, \nabla \cdot \chi_h) = \left(f^{n+\frac{1}{2}}, \chi_h\right), \\ \tilde{u}_h^{n+1} = 0 \text{ on } \partial\Omega. \end{cases} \tag{3.11}$$

Step 1 RotProjB-SKEW is the same as Step 1 RotProjB-EMAC except the nonlinear term.

Step 2: Solve for ϕ^{n+1}

$$\begin{cases} \frac{-\nabla \cdot \tilde{u}_h^{n+1}}{\Delta t} + \Delta \phi_h^{n+1} = 0, \\ u_h^{n+1} \cdot n = 0 \text{ on } \partial\Omega. \end{cases} \tag{3.12}$$

From here, we first recover p_h^{n+1} from $\phi_h^{n+1} = \frac{1}{2}\left(p_h^{n+1} - p_h^n + \nu \nabla \cdot \tilde{u}_h^{n+1}\right)$, and then recover u_h^{n+1} from $\tilde{u}_h^{n+1}$ and p_h^{n+1} using the projection [41]. This formulation is a second order method from (3.6)-(3.8) in [41], equipped with the EMAC nonlinearity formulation.

3.3.1 Planar Lattice Flow

We first consider an investigation of the evolution of an initial velocity and flow of four vortices which are rotating opposite to one another. This particular phenomenon, named “planar lattice flow”, is a solution to the stationary incompressible Euler equation and has been studied in detail in [81, 80, 79]. Let

$x \in \Omega = (0,1)^2$ and define the initial velocity and true velocity/pressure pair (u, p) as

$$u_0(x) = \begin{bmatrix} \sin(2\pi x_1)\sin(2\pi x_2) \\ \cos(2\pi x_1)\cos(2\pi x_2) \end{bmatrix},$$
$$u(t,x) = u_0(x)e^{-8\pi^2 \nu t},$$
$$p(t,x) = \frac{1}{4}\left[\cos(4\pi x_1) - \cos(4\pi x_2)\right] e^{-16\pi^2 \nu t}.$$

Periodic boundary conditions are imposed on $\partial\Omega$ and we enforce the integral zero-mean condition on the pressure. We do not impose an external force, so $f = 0$, and we set $\nu = 4 \times 10^{-6}$. Crank-Nicolson time stepping was used for the coupled schemes (not including SKEW-BE-PROJ and EMAC-BE-PROJ, which use Backward Euler), and we set $\Delta t = .001$ with the end time $T = 5$ for all methods. A uniform mesh with (P_2, P_1) Taylor hood elements was used with a mesh width of $h = \frac{1}{48}$. Figure 3.1 depicts the initial velocity of the problem.

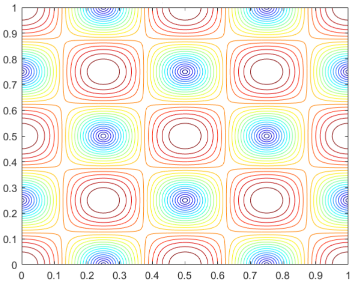

Figure 3.1: Shown above is the initial velocity u_0 for planar lattice flow.

The true solution of this problem is for it to decay exponentially with time, but to remain stationary in space. As time goes on, the term $e^{-8\pi^2\nu t}$ will uniformly decay the initial condition. Plots of computed solutions at t=5 are shown in figure 3.2. For each formulation using the SKEW nonlinear term, we observe oscillations to the point where do not see anything that resembles the correct solution. However, the EMAC formulations strongly resemble Figure 3.1, although some error is clearly present.

Figure 3.3 shows a semilog plot of L^2 error at every timestep for each of the six methods, and as expected we see much better performance from the EMAC methods over the SKEW methods. Specifically, we notice nearly identical error for EMAC and RotProjB-EMAC, where EMAC-BE-PROJ performs slightly worse. This is expected because it is a first order method, so it naturally will not outperform EMAC and RotProjB-EMAC. There does not seem to be a huge difference in L^2 error otherwise. The SKEW methods do not perform well at all over time, and we observe a very large L^2 error which level off at $O(10)$ only due to the stability of the method.

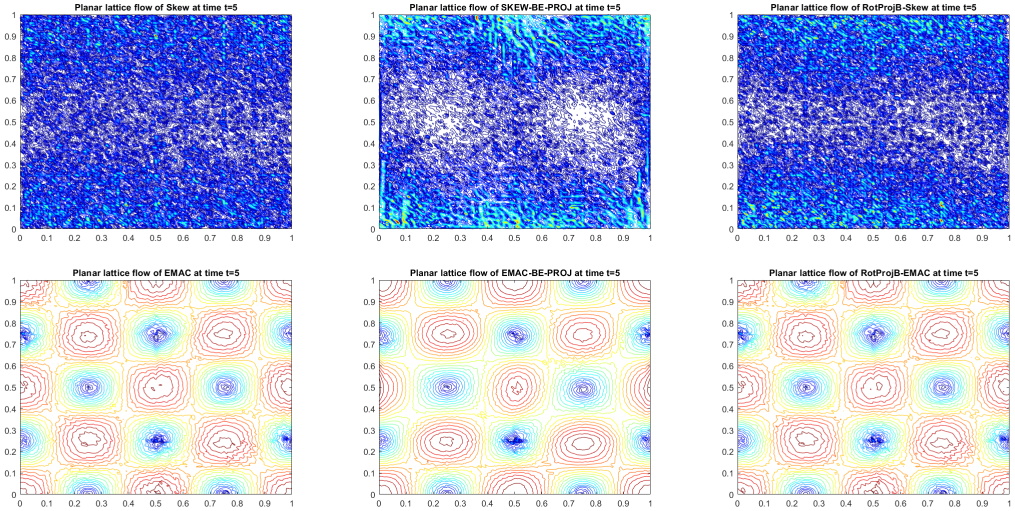

Figure 3.2: Plots of the solution of each formulation at time $t = 5$

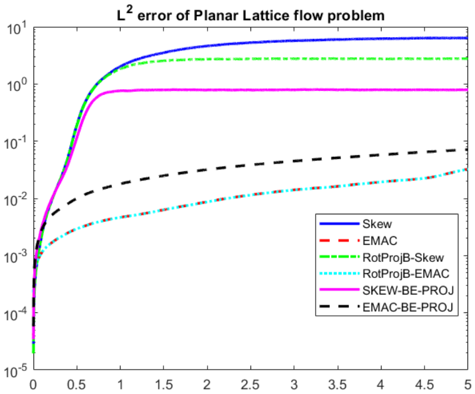

Figure 3.3: Shown above L^2 error for each formulation vs. time for the planar lattice vortex problem.

3.3.2 Gresho problem

We next test the methods on the Gresho standing vortex problems. Computations involving coupled schemes using SKEW and EMAC are well documented [19, 68], but we include them here with projection method results for the sake of comparison. The initial velocity is a solution to the steady Euler equations, so it is stationary in time, which makes it easy to measure the effectiveness of a formulation. It will also help measure how effective a method is in conservation properties, since we assume $f = 0$ and $\nu = 0$.

We begin by recalling the Gresho problem from Section 4.2.2, where we define $r = \sqrt{x^2 + y^2}$ on $\Omega = (-0.5, 0.5)^2$. The velocity and pressure are defined as

$$u = \begin{cases} \begin{bmatrix} -5y \\ 5x \end{bmatrix} & \text{for } r < 2, \\ \begin{bmatrix} \frac{2y}{r} + 5y \\ \frac{2x}{r} - 5x \end{bmatrix} & \text{for } .2 \le r \le .4, \\ \begin{bmatrix} 0 \\ 0 \end{bmatrix} & \text{for } r > .4, \end{cases}$$

$$p = \begin{cases} 12.5r^2 + C_1 & \text{for } r < .2, \\ 12.5r^2 - 20r + 4\log(r) + C_2 & \text{for } .2 \le r \le .4, \\ 0 & \text{for } r > .4, \end{cases}$$

where

$$\begin{aligned} C_2 &= -12.5(.4)^2 + 20(.4)^2 - 4\log(.4), \\ C_1 &= C_2 - 20(.2) + 4\log(.2). \end{aligned}$$

We again use Crank-Nicolson time stepping for the coupled schemes and Backward Euler for SKEW-BE-PROJ and EMAC-BE-PROJ. We solve using $\Delta t = .01$ and set $T = 4$. Taylor-Hood (P_2, P_1) elements were used with a mesh size of $h = \frac{1}{48}$. Similar to the planar lattice flow problem in the previous section, we have an initial velocity (shown in figure 3.4) that we compare to the computed solutions at later times. The difference is that the vortex should maintain its shape because it is a solution to a steady-state problem. However, numerical errors are inevitable, and accumulate after many iterations. Previous work in [19, 51] shows that EMAC tends to outperform most conventional formulations over time, and we will again observe this phenomena for this test problem.

The initial velocity is shown in figure 3.4, and computed solutions at $t = 1, 2, 3, 4$ are shown in figure 4.2. SKEW, SKEW-BE-PROJ, and RotProjB-SKEW produce very poor solutions, as the velocity has dissipated very rapidly compared to the EMAC solutions. Coupled EMAC, EMAC-BE-PROJ, and RotProjB-EMAC do a better (although not great) job at maintaining the vortex for longer times, with the

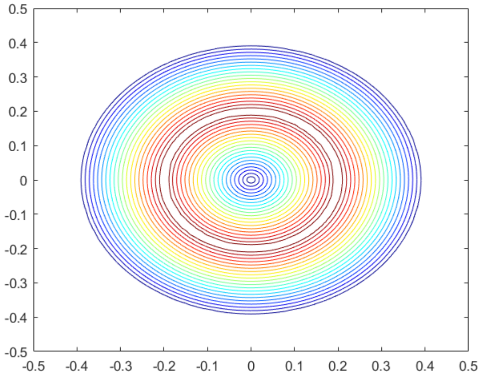

Figure 3.4: Initial velocity for the Gresho problem.

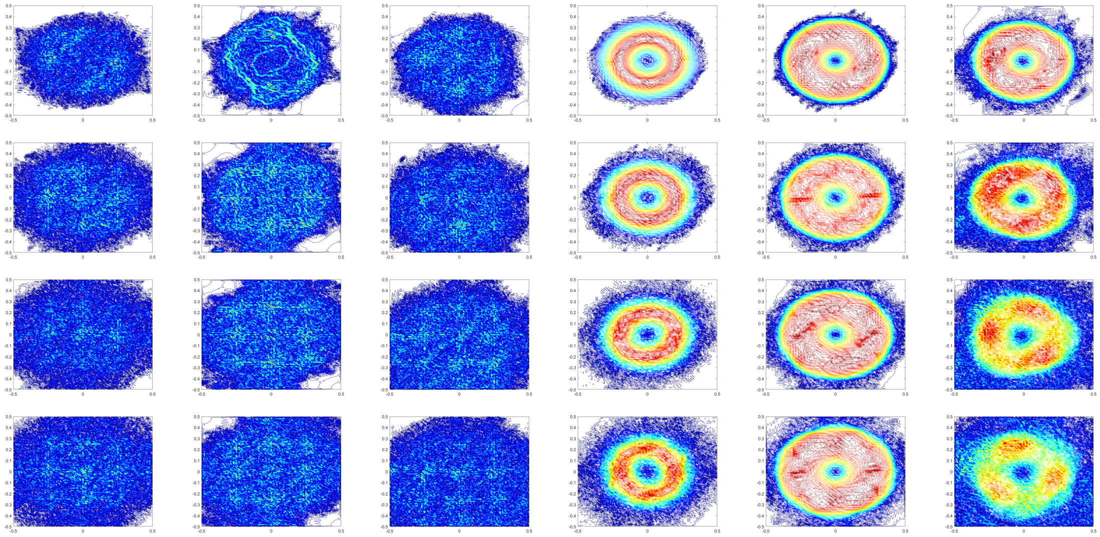

Figure 3.5: Numerical results of SKEW coupled, SKEW-BE-PROJ, RotProjB-SKEW, EMAC coupled, EMAC-BE-PROJ, RotProjB-EMAC (from left to right, respectively) at times $t = 1, 2, 3, 4$ (top to bottom).

coupled EMAC outperforming the other EMAC methods, especially at later times.

Additionally, we calculated L^2 error, energy, momentum, and angular momentum for each method. Coupled EMAC is the only one to conserve energy, with coupled SKEW producing slight dissipation (the severe oscillations in coupled SKEW were enough to cause it to slightly dissipate energy after $t = 0.5$). All projection methods significantly dissipate energy, nonphysically reducing it by nearly 50% by the end time. For momentum, the coupled schemes, SKEW-BE-PROJ, and EMAC-BE-PROJ maintain a constant momentum while both projection methods had slight decreases. We expect this behavior for the four formulations that conserved it, as that is what the theory suggests. The RotProjB methods both do not have formal conservation analysis done, so we did not specifically expect conservation.

For angular momentum, we do not observe conservation for any of the SKEW methods, which is not surprising. However we see conservation for EMAC, which is of course expected. If one takes a closer look at the bottom right hand corner of the angular momentum plot, they would notice some small bumps in the angular momentum for RotProjB-EMAC. Once again, we do not have any conservation results on this formulation, but it does not seem like angular momentum is necessarily conserved here. However, we see a slight increase in angular momentum for EMAC-BE-PROJ. We have angular momentum conservation in theory, however coupled EMAC is second order convergent in time and our projection method is half order convergent. This increase in angular momentum is likely a result of pollution due to larger errors with respect to the time steps. Lastly, the L^2 error plot shows the EMAC schemes dramatically outperformed the SKEW schemes, with coupled EMAC beating out RotProjB-EMAC and SKEW-BE-PROJ. Interestingly enough, the L^2 error for EMAC-BE-PROJ starts worse off than RotProjB-EMAC, but at around $t = 2.5$ it is smaller.

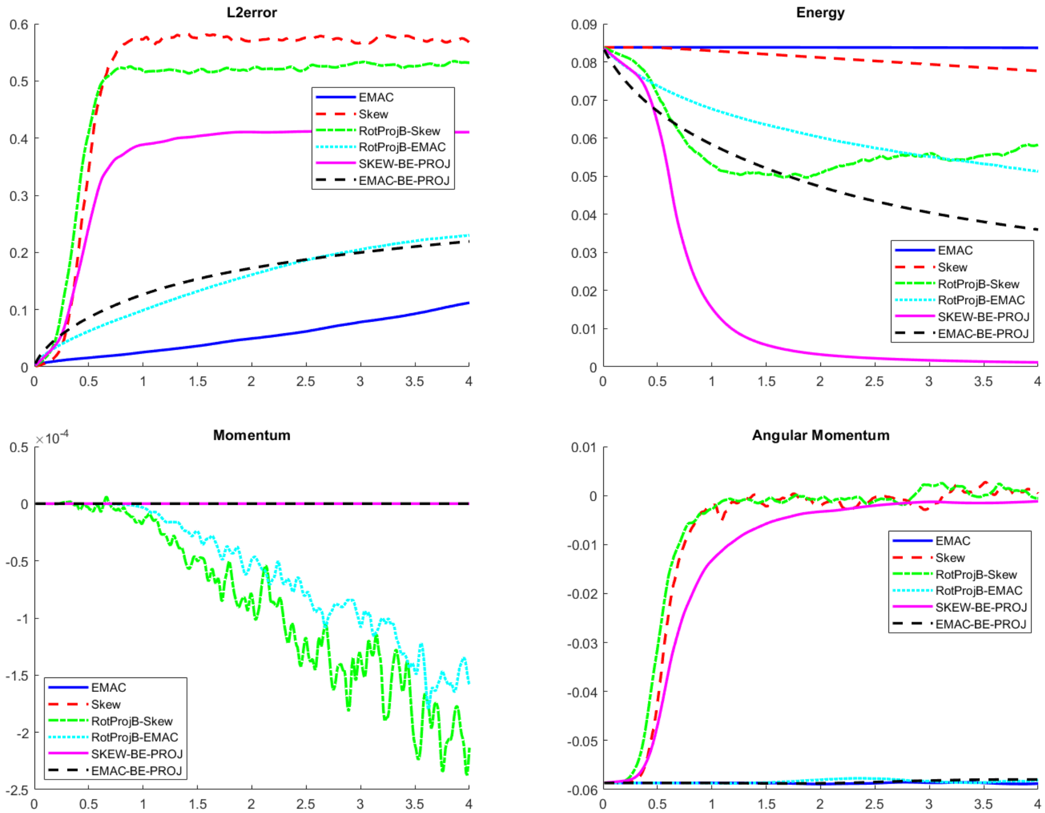

Figure 3.6: L^2 error, energy, momentum, and angular momentum plots of SKEW, EMAC, SKEW-BE-PROJ, EMAC-BE-PROJ, RotProjB-SKEW, and RotProjB-EMAC.

3.3.3 Contaminant Flow Analysis

For our last test, we consider EMAC and SKEW coupled schemes for prediction of river contamination. We chose the three rivers in Pittsburgh PA (USA) where they meet; in Figure 3.7, observe the northeastern river (Allegheny River) and the southeastern river (Monongahela River) meet to form the Ohio river. The contaminant is modeled with the fluid transport equation

$$c_t + u \cdot \nabla c - \varepsilon \Delta c = 0,$$

where c is the contaminant, u is the velocity, and ε is the diffusion coefficient.

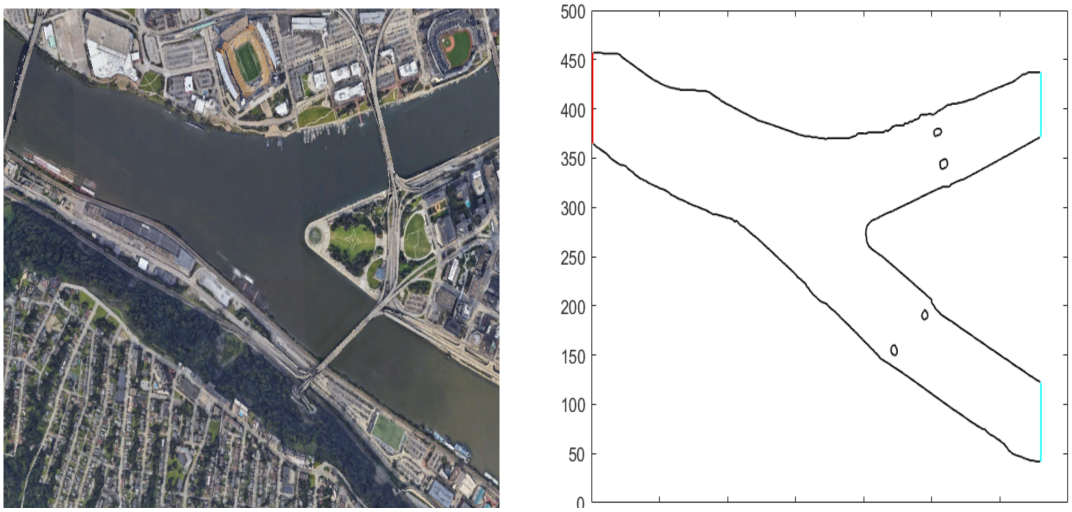

Figure 3.7: Satellite image of the rivers in Pittsburgh, PA (left) and mesh outline used in computation (right).

A domain is created (shown in figure 3.7) by converting the image to grayscale via the isoline package in FreeFEM. The bridges were pre-edited out, and were replaced with pylons represented by circles. The domain is such that there are two inlets and one outlet, where the two inlets are the Allegheny and Monongahela Rivers and the outlet is the Ohio River. The inlets are colored cyan on the right hand side and the outlet is colored red.

We used the Stokes solution for the initial condition, BDF2 timestepping with $\Delta t = .01$ together with Taylor-Hood (P_2, P_1) elements , $Re = \nu^{-1} = 100$, and an end time of $T = 15$. Additionally, we used grad-div stabilization with $\gamma = 1$ and Newton iterations to solve the nonlinear problem. For both EMAC and SKEW we used a constant inflow of $u = \begin{bmatrix} -20, 0 \end{bmatrix}^T$ on the Monongahela and Allegheny inflows as well as do-nothing outflow on the Ohio River.

For the contaminant flow, we used an initial condition of

$$c = \begin{cases} 1, & \text{if } (x-567)^2 + (y-371)^2 < 25^2, \\ 1, & \text{if } (x-567)^2 + (y-131)^2 < 25^2, \\ 0, & \text{otherwise.} \end{cases}$$

This gives 2 circles at the same x coordinate on both the Monongahela and Allegheny rivers. We took $\epsilon = .001$, used P_2 elements, BDF2 timestepping and $\Delta t = 0.01$. Lastly, we had do-nothing boundary conditions for each boundary except the inflows, which were set to zero.

We ran simulations with 112229 total degrees of freedom to compare SKEW and EMAC, and also computed a reference solution using 249162 degrees of freedom and the convective nonlinear term, up to time $t = 15$. The solution at $t = 15$ is shown in figure 3.8. EMAC and SKEW solutions are shown in plots at times $t = 1$, 9, and 15 in figure 3.9; we observe that the SKEW solution has significant oscillations which destroy its solution, while the EMAC solution remains stable and exhibits only minor oscillations. Similarly with the concentrations shown in figure 3.10, where EMAC is stable and matches the resolved solution qualitatively well but SKEW's solution is killed by oscillations.

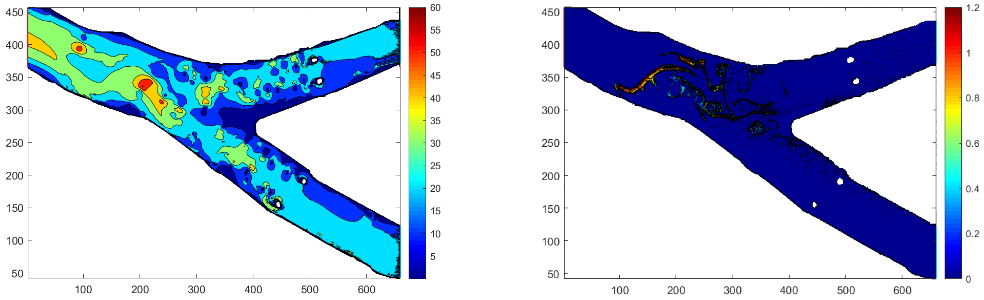

Figure 3.8: Reference velocity (left) and contaminant flow (right) at time $t = 15$.

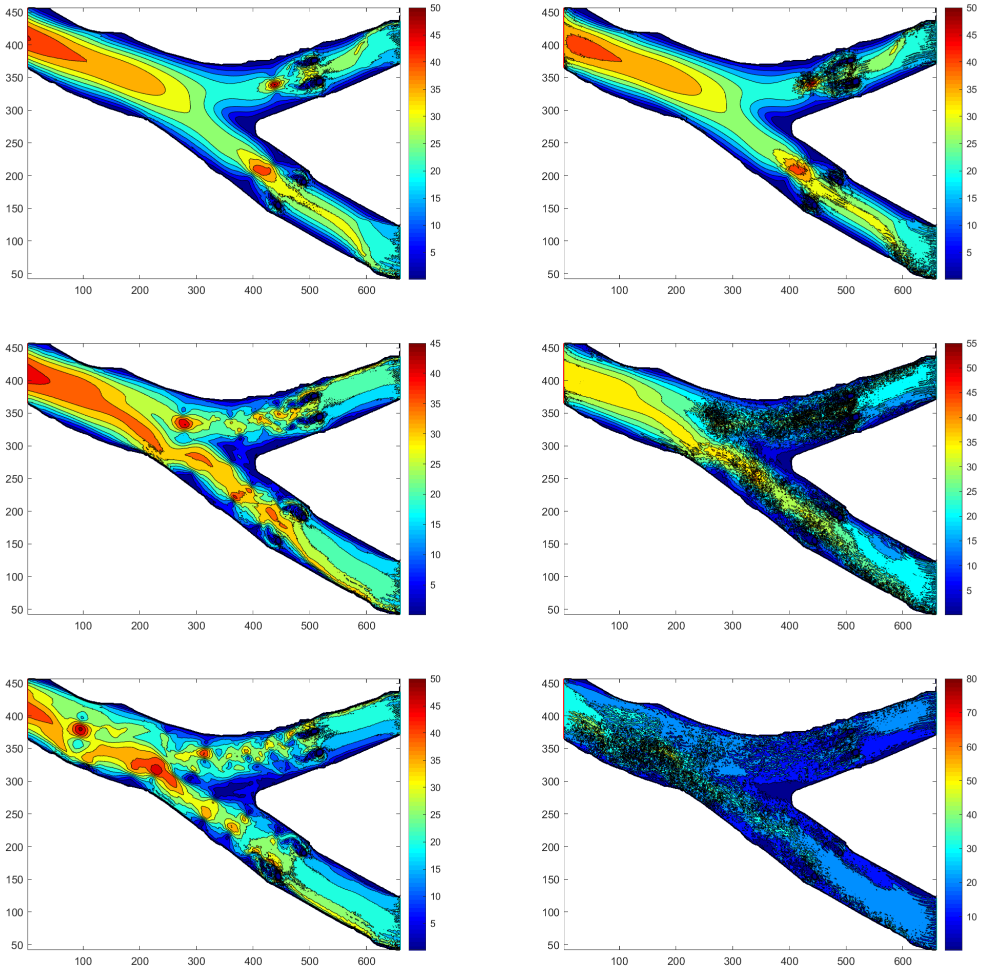

Figure 3.9: Velocity at times $t = 3$, 9, and 15 for EMAC (left) versus SKEW (right).

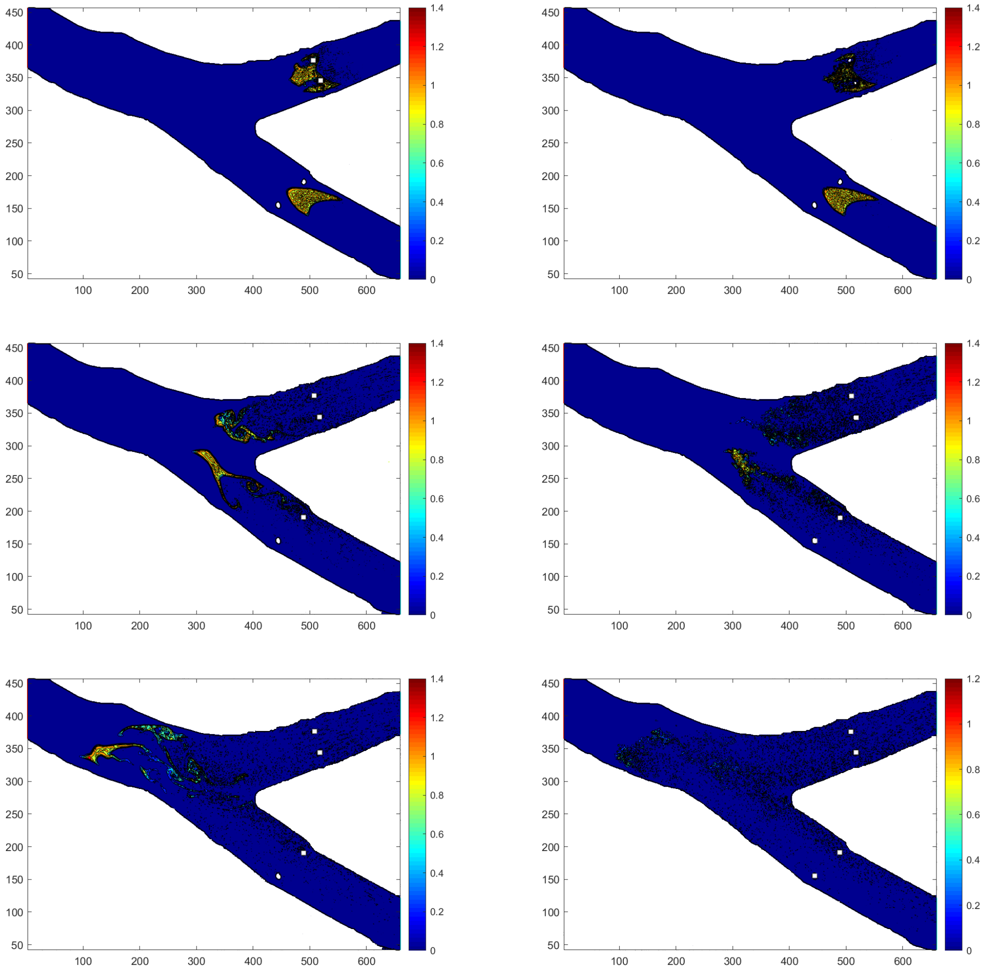

Figure 3.10: Contaminant flow at times $t = 3$, 9, and 15 for EMAC (left) versus SKEW (right).

Chapter 4

EMAC-Reg

We have introduced in Chapters 1 and 2 regularization models and quantity conserving models for the NSE. Regularization models are appealing because they allow us to run experiments on coarser meshes, however one of the trade-offs is that they do not conserve quantities like energy, momentum, and angular momentum. Our goal is to have a regularization model that conserves all of these quantities, so we apply the EMAC nonlinear term to a particular regularization model to achieve this goal.

In this chapter, we introduce the energy, momentum, and angular momentum conserving regularization formulation (EMAC-Reg), show that it is a well-posed problem, perform a semi-discrete error analysis, and perform numerical tests. We also show that, like EMAC, EMAC-Reg conserves energy, momentum, and angular momentum. For our numerical tests, we perform a convergence analysis for both h and Δt. We test the standing vortex problem, or the Gresho problem, and compare the speed contours, L^2 error, and other quantities of other formulations. We do the same for flow past a step and Kelvin-Helmholtz instability.

4.1 The EMAC-Reg scheme and its analysis

We consider the following semi-discretization of EMAC-Reg: Given $w_h^0, u_h^0 \in V_h$, find $(u_h, P_h, w_h, \lambda_h) \in (X_h, Q_h, X_h, Q_h) \times (0, T]$ for all $(v_h, q_h, \chi_h, r_h) \in (X_h, Q_h, X_h, Q_h)$,

$$((u_h)_t, v_h) + c(w_h, w_h, v_h) - (P_h, \nabla \cdot v_h) + \nu (\nabla u_h, \nabla v_h) = (f, v_h), \tag{4.1}$$

$$(\nabla \cdot u_h, q_h) = 0, \tag{4.2}$$

$$(\lambda_h, \nabla \cdot \chi_h) + \alpha^2 (\nabla w_h, \nabla \chi_h) + (w_h, \chi_h) = (u_h, \chi_h), \tag{4.3}$$

$$(\nabla \cdot w_h, r_h) = 0. \tag{4.4}$$

Even though in the continuous model (1.2)-(1.4) the constraint $\nabla \cdot u = 0$ follows from the model definition, in a discretization it does not. Hence we enforce that the velocity be discretely divergence-free with (4.2), and thus add the Lagrange multiplier to (4.3) so the system is not overdetermined.

The analysis below is for the semi-discrete system (4.1)-(4.4). Extensions to particular timestepping methods can be done, and for A-stable methods such as backward Euler, BDF2, or Crank-Nicolson, analogous stability results can be obtained (with expected additional error depending on time step size). For conservation properties, only Crank-Nicolson will allow for exact conservation; other timestepping methods' discrete time derivative term will generally contribute to the energy, momentum, and angular momentum balances. It is believed, with some evidence from [19, 20] that this effect is much less problematic than the nonlinear term contributing to these balances. While it is discussed in [23] that Crank-Nicolson will provide optimal convergence in space and time, whether an improved Gronwall constant can be obtained is an open problem.

4.1.1 Stability and well-posedness

We first prove stability of the scheme, followed by well-posedness.

Theorem 9. *Let $(u_h, P_h, w_h, \lambda_h)$ solve (4.1)-(4.4) with $f \in L^2(0, T; X')$ and $w_h^0 \in H^1(\Omega)$. Additionally,* *assume the LBB condition is satisfied on this finite element space. The following bounds hold:*

$$\begin{aligned} ||w_h(T)||^2 + \alpha^2 ||\nabla w_h(T)||^2 + \nu \left[||\nabla w_h||^2_{L^2(0,T;L^2)} + \alpha^2 ||A_h w_h||^2_{L^2(0,T;L^2)} \right] & \\ \leq \nu^{-1} ||f||^2_{L^2(0,T;X')} + ||w_h(0)||^2 + \alpha^2 ||\nabla w_h(0)||^2 , & \qquad (4.5) \\ ||u_h||^2_{L^2(0,T;L^2)} \leq (T + \alpha^2)\nu^{-1} ||f||^2_{L^2(0,T;X')} + ||w_h(0)||^2 & \\ + \alpha^2 ||\nabla w_h(0)||^2 + ||u_h(0)||^2 . & \end{aligned}$$

Proof. Choosing $v_h = w_h$ in (4.1) gives us

$$((u_h)_t, w_h) + \nu (\nabla u_h, \nabla w_h) = (f, w_h) , \tag{4.6}$$

where the trilinear term $c(w_h, w_h, w_h) = 0$ and $(P_h, \nabla \cdot w_h) = 0$ by (4.4).

Next choose $\chi_h = A_h w_h$ in (4.3) and then $\chi_h = w_h$ in (4.3) to obtain

$$\alpha^2 ||A_h w_h||^2 + ||\nabla w_h||^2 = (\nabla u_h, \nabla w_h) , \tag{4.7}$$

$$\alpha^2 ||\nabla w_h||^2 + ||w_h||^2 = (u_h, w_h) . \tag{4.8}$$

To find another expression for $((u_h)_t, w_h)$, we take the temporal derivative of (4.8) to get

$$\alpha^2 \frac{d}{dt} ||\nabla w_h||^2 + \frac{d}{dt} ||w_h||^2 = \frac{d}{dt} (u_h, w_h) = ((u_h)_t, w_h) + (u_h, (w_h)_t) ,$$

and so

$$((u_h)_t, w_h) = \frac{d}{dt} \left[||w_h||^2 + \alpha^2 ||\nabla w_h||^2 \right] - (u_h, (w_h)_t) . \tag{4.9}$$

Choosing $\chi_h = (w_h)_t$ in (4.3) yields

$$\frac{\alpha^2}{2} \frac{d}{dt} ||\nabla w||^2 + \frac{1}{2} \frac{d}{dt} ||w_h||^2 = (u_h, (w_h)_t) , \tag{4.10}$$

and now combining (4.9) and (4.10) provides

$$\begin{aligned} ((u_h)_t, w_h) &= \frac{d}{dt}\left[||w_h||^2 + \alpha^2 ||\nabla w_h||^2 \right] - \frac{d}{dt}\left[\frac{\alpha^2}{2} ||\nabla w_h||^2 + \frac{1}{2} ||w_h||^2 \right] \\ &= \frac{1}{2}\frac{d}{dt}\left[||w_h||^2 + \alpha^2 ||\nabla w_h||^2 \right]. \end{aligned} \tag{4.11}$$

Using (4.7) and (4.11) in (4.6) gives us

$$\frac{1}{2}\frac{d}{dt}\left[||w_h||^2 + \alpha^2 ||\nabla w_h||^2 \right] + \nu \left[\left||\nabla w_h^2\right|| + \alpha^2 ||A_h w_h||^2 \right] = (f, w_h). \tag{4.12}$$

We bound the right hand side of (4.12) using Cauchy-Schwarz and Young's inequalities,

$$(f, w_h) \leq ||f||_{X'} ||w_h|| \leq \frac{\nu}{2} ||\nabla w_h||^2 + \frac{1}{2\nu} ||f||_{X'}^2,$$

and after inserting this bound into (4.12) and then integrating in time gives us the first stated estimate

$$\begin{aligned} &||w_h(T)||^2 + \alpha^2 ||\nabla w_h(T)||^2 + \nu \left[||\nabla w_h||_{L^2(0,T;L^2)}^2 + \alpha^2 ||A_h w_h||_{L^2(0,T;L^2)}^2 \right] \\ \leq &\nu^{-1} ||f||_{L^2(0,T;X')}^2 + ||w_h(0)||^2 + \alpha^2 ||\nabla w_h(0)||^2. \end{aligned}$$

To get the second bound, we test (4.3) with $\chi_h = u_h$ to give us

$$||u_h||^2 = \alpha^2 (\nabla w_h, \nabla u_h) + (w_h, u_h). \tag{4.13}$$

Applying (4.7) and (4.8), (4.13) becomes

$$||u_h||^2 = \alpha^2 \left[\alpha^2 ||A_h w_h||^2 + ||\nabla w_h||^2 \right] + \alpha^2 ||\nabla w_h||^2 + ||w_h||^2. \tag{4.14}$$

Now using (4.5) in (4.14) and integrating with respect to time, we get

$$\begin{aligned} ||u_h||_{L^2(0,T;L^2)}^2 \leq &\alpha^2 \left[\alpha^2 ||A_h w_h||_{L^2(0,T;L^2)}^2 + ||\nabla w_h||_{L^2(0,T;L^2)}^2 \right] + T\nu^{-1} ||f||_{L^2(0,T;X')}^2 \\ &+ ||w_h(0)||^2 + \alpha^2 ||\nabla w_h(0)||^2 + ||u_h(0)||^2. \end{aligned}$$

Finally using (4.5) to bound the the quantity in brackets yields the second result of the theorem:

$$\begin{aligned} ||u_h||^2_{L^2(0,T;L^2)} \leq &(T+\alpha^2)\nu^{-1} ||f||^2_{L^2(0,T;X')} + ||w_h(0)||^2 \\ &+ \alpha^2 ||\nabla w_h(0)||^2 + ||u_h(0)||^2 . \end{aligned}$$

■

Theorem 10. *The semidiscrete EMAC-Reg system* (4.1)-(4.4) *is well-posed.*

Proof. The nonlinearity of the system of ODEs defining w_h and u_h is quadratic, and thus locally Lipschitz. By classical ODE theory we may conclude that a solution exists and is unique so long as all possible solutions cannot blow up in finite time [61], which is established in Theorem 3.1 and due to the LBB assumption on the finite element space it is true for P_h and λ_h. ■

4.1.2 Conservation of energy, momentum, and angular momentum

For NSE (Lagrangian mechanics in general), energy, momentum, and angular momentum are defined by

$$\begin{aligned} \text{Kinetic energy} \quad & E = \frac{1}{2}(u,u) = \frac{1}{2}\int_\Omega |u|^2 dx, \\ \text{Linear momentum} \quad & M = \int_\Omega u\, dx, \\ \text{Angular momentum} \quad & M_x = \int_\Omega u \times x\, dx. \end{aligned}$$

Let $e_1 = (1,0,0)^T$, $e_2 = (0,1,0)^T$, $e_3 = (0,0,1)^T$ and $\phi_i = x \times e_i$ for $i = 1,2,3$. The definition of momentum and angular momentum can now be equivalently written as:

$$\begin{aligned} M_i &= \int_\Omega u_i\, dx = (u, e_i), \\ (M_x)_i &= \int_\Omega (u \times x) \cdot e_i\, dx = (u, \phi_i), \text{ for } i = 1,2,3. \end{aligned}$$

Conservation laws for models of fluid flow are determined by the model itself, and in some sense this should be analogous to the true physical law. For EMAC-Reg, we prove below that the model energy, momentum, and angular momentum conserved quantities are given by

$$
\begin{aligned}
E_{\text{EMAC-Reg}} &:= \frac{1}{2}\int_\Omega u \cdot w \, dx = \frac{1}{2}\left(u, w\right) \, = \frac{1}{2}\left|\left|w\right|\right|^2 + \frac{\alpha^2}{2}\left|\left|\nabla w\right|\right|^2 , \\
M_{\text{EMAC-Reg}} &:= \int_\Omega u \, dx = \left(u, e_i\right) , \\
AM_{\text{EMAC-Reg}} &:= \int_\Omega u \times x \, dx = \left(u, \phi_i\right) .
\end{aligned}
$$

For regularization models, altered quantities are generally conserved on the continuous level due to the regularization's modification of the physical equation. *Our goal is that conserved quantities in our discretization matches that of the continuous model.*

Following [19], for our conservation law study, we notice that practical boundary conditions will alter the balance of the quantities above. Enforcing conditions strongly or weakly will also impact the balance for the numerical experiments. It is for these reasons that we isolate the affect on the treatment of the nonlinearity of the quantities from the boundary conditions. We assume for the numerical experiments in section 4 that the finite element solution $(u_h, P_h, w_h, \lambda_h)$ is supported in a compact subdomain $\hat{\Omega} \subsetneq \Omega$. Not coincidentally, the conserved energy is the same as is conserved by the related NS-α model and the ADM turbulence model of Stolz and Adams [63].

Theorem 11. *Assume that the finite element solutions u_h and p_h are supported in a compact subdomain $\hat{\Omega} \subsetneq \Omega$ so that there exists a strip $S = \Omega \backslash \hat{\Omega}$ along $\partial\Omega$ where u_h vanishes. This is also assumed for the forcing function f. Then the EMAC-Reg formulation conserves energy, momentum, and angular momentum for $\nu = 0,\ f = 0$. In other words,*

$$
\begin{aligned}
&E_{EMAC\text{-}Reg}(t) = E_{EMAC\text{-}Reg}(0), \\
&M_{EMAC\text{-}Reg}(t) = M_{EMAC\text{-}Reg}(0), \\
&AM_{EMAC\text{-}Reg}(t) = AM_{EMAC\text{-}Reg}(0),
\end{aligned}
$$

for all t.

Proof. We start by showing energy conservation. We test (4.1) with $v = w_h$,

$$
\left((u_h)_t, w_h\right) \, + c(w_h, w_h, w_h) + \nu\left(\nabla u_h, \nabla w_h\right) \, = \left(f, w_h\right) .
$$

Since $c(w_h, w_h, w_h) = 0$ and we assumed $f = 0$ and $\nu = 0$, we are left with

$$((u_h)_t, w_h) = 0 \Rightarrow \frac{d}{dt}(u_h, w_h) = 0,$$

and hence energy is preserved.

For momentum conservation, we cannot simply test (4.1) with e_i, because it is not in the finite element space X_h. To remedy this, we recall that $u_h = 0$ on the strip S. Define $\chi(g) \in X_h$ for $g = e_i$ and ϕ_i such that $\chi(g) = g$ in $\hat{\Omega}$ and arbitrary on $S = \Omega \backslash \hat{\Omega}$ to satisfy the boundary conditions of X_h.

For momentum conservation we test (4.1) with $\chi(e_i)$. This gives

$$((u_h)_t, e_i) + c(w_h, w_h, e_i) + \nu (\nabla u_h, \nabla e_i) = (f, e_i).$$

For the nonlinear term

$$\begin{aligned} c(w_h, w_h, e_i) &= ((\nabla w_h) w_h, e_i) + \left((\nabla w_h)^T w_h, e_i\right) + ((\nabla \cdot w_h) w_h, e_i) \\ &= (w_h \cdot \nabla w_h, e_i) + (w_h, (\nabla w_h) e_i) + ((\nabla \cdot w_h) w_h, e_i), \end{aligned}$$

where we get the above identity from (2.9). Then applying (2.7) and (2.8) to the first term and using that $\nabla e_i = 0$ gives

$$\begin{aligned} c(w_h, w_h, e_i) &= -(w_h \cdot \nabla e_i, w_h) - ((\nabla \cdot w_h) w_h, e_i) + (e_i \cdot \nabla w_h, w_h) + ((\nabla \cdot w_h) w_h, e_i) \\ &= (e_i \cdot \nabla w_h, w_h) \\ &= -\frac{1}{2}((\nabla \cdot e_i) w_h, w_h) = 0. \end{aligned}$$

Combining this with $f = \nu = 0$, the fact that $\nabla \cdot e_i = 0$ and that e_i doesn't depend on t, we end up with

$$\frac{d}{dt}(u_h, e_i) = 0,$$

which is precisely momentum conservation.

For angular momentum, we test (4.1) with $\chi(\phi_i)$, which gives

$$((u_h)_t, \phi_i) + c(w_h, w_h, \phi_i) + \nu (\nabla u_h, \nabla \phi_i) = (f, \phi_i).$$

We begin by investigating $\nabla \phi_i$. First let $x = \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix}^T$, then ϕ_i for $i = 1, 2, 3$ takes the form

$$\phi_1 = \begin{bmatrix} 0 \\ x_3 \\ -x_2 \end{bmatrix}, \quad \phi_2 = \begin{bmatrix} -x_3 \\ 0 \\ x_1 \end{bmatrix}, \quad \phi_3 = \begin{bmatrix} x_2 \\ -x_1 \\ 0 \end{bmatrix}.$$

This immediately gives

$$\nabla\phi_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}, \quad \nabla\phi_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}, \quad \nabla\phi_3 = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}. \tag{4.15}$$

With this in mind, we now investigate the nonlinear term and apply (4.15) along with $\nabla \cdot \phi_i = 0$ to get

$$\begin{aligned} c(w_h, w_h, \phi_i) &= -(w_h \cdot \nabla\phi_i, w_h) - ((\nabla \cdot w_h) w_h, \phi) + (\phi_i \cdot \nabla w_h, w_h) + ((\nabla \cdot w_h) w_h, \phi_i) \\ &= -(w_h \cdot \nabla\phi_i, w_h) - \frac{1}{2}((\nabla \cdot \phi_i) w_h, w_h) \\ &= 0. \end{aligned}$$

Combining this with $f = \nu = 0$ provides

$$\frac{d}{dt}(w_h, \phi_i) = 0,$$

which is angular momentum conservation. ■

4.1.3 Error analysis

In this subsection, we provide the error analysis for EMAC-Reg. The discretized formulation is subtracted from the weak formulation and we manipulate the nonlinear term to get an appropriate error bound. Below is the result.

Theorem 12. *Let $(u_h, P_h, w_h, \lambda_h)$ solve (4.1)-(4.4) and (u, p, w) solve (1.2)-(1.4) with an EMAC nonlinear term, $w_t \in L^2(0, T; X')$, $w, \nabla w \in L^1(0, T; L^\infty)$ and $P \in L^2(0, T; L^2)$. Denote $e^w(t) = w(t) - w_h(t) = \phi_h^w + \eta^w$ where $\eta^w = w - I_h^{St}(w)$ and $\phi_h^w = I_h^{St}(w) - w_h$. Define e^u, η^u, ϕ_h^u similarly. We get the following error estimates:*

$$
\begin{aligned}
&\alpha^2 \left|\left|\nabla e^w(T)\right|\right|^2 + \left|\left|e^w(T)\right|\right|^2 + \nu \int_0^T \left[2\alpha^2 \left|\left|A_h e^w\right|\right|^2 + \left|\left|\nabla e^w\right|\right|^2\right] dt \\
\leq &\alpha^2 \left|\left|\nabla \eta^w(0)\right|\right|^2 + \left|\left|\eta^w(0)\right|\right|^2 + \nu \left[2\alpha^2 \left|\left|A_h \eta^w\right|\right|^2_{L^2(0,T;L^2)} + \left|\left|\nabla \eta^w\right|\right|^2_{L^2(0,T;L^2)}\right] \\
&+ K \left[\alpha^2 \left|\left|\nabla \phi_h^w(0)\right|\right|^2 + \left|\left|\phi_h^w(0)\right|\right|^2 + 4\nu^{-1} \left(\left|\left|\eta_t^w\right|\right|^2_{L^2(0,T;X')} + \inf_{q_h \in L^2(0,T;Q_h)} \left|\left|P - q_h\right|\right|^2_{L^2(0,T;L^2)}\right)\right].,
\end{aligned}
$$

where

$$
K = \exp\left[2C\left(\left|\left|w\right|\right|_{L^1(0,T;L^\infty)} + \left|\left|\nabla w\right|\right|_{L^1(0,T;L^\infty)}\right) + T\nu\right],
$$

and

$$
\begin{aligned}
&\left|\left|e^u(T)\right|\right|^2 \\
\leq &\alpha^2 \left|\left|\nabla \eta^w(0)\right|\right|^2 + \left|\left|\eta^w(0)\right|\right|^2 + \nu \left[2\alpha^2 \left|\left|A_h \eta^w\right|\right|^2_{L^2(0,T;L^2)} + \left|\left|\nabla \eta^w\right|\right|^2_{L^2(0,T;L^2)}\right] \\
&+ K \left[\alpha^2 \left|\left|\nabla \phi_h^w(0)\right|\right|^2 + \left|\left|\phi_h^w(0)\right|\right|^2 + 4\nu^{-1} \left(\left|\left|\eta_t^w\right|\right|^2_{L^2(0,T;X')} + \inf_{q_h \in L^2(0,T;Q_h)} \left|\left|P - q_h\right|\right|^2_{L^2(0,T;L^2)}\right)\right] + 3\left|\left|\eta^u\right|\right|^2.
\end{aligned}
$$

Corollary 13. *Let $(u_h, P_h, w_h, \lambda_h)$ solve (4.1)-(4.4) and (u, p, w) satisfy the same criteria as in Theorem 12. Additionally suppose $u, w \in L^\infty(0, T; H^{k+1})$, $w_t \in L^2(0, T; H^{k+1})$, and $P \in L^2(0, T; H^k)$. Assuming $X_h = X \cap P_k(\mathcal{T}_k)$ and $Q_h = Q \cap P_{k-1}(\mathcal{T}_k)$, where P_k denotes the set of degree k polynomials, satisfies the infsup condition, we get from (2.4)*

$$
\begin{aligned}
&\alpha^2 \left|\left|\nabla e^w(T)\right|\right|^2 + \left|\left|e^w(T)\right|\right|^2 + \nu \int_0^T \left[2\alpha^2 \left|\left|A_h e^w\right|\right|^2 + \left|\left|\nabla e^w\right|\right|^2\right] dt \\
\leq &C\left(\alpha^2 h^{2k} |w(0)|^2_{k+1} + h^{2k+2} |w(0)|^2_{k+1} + \nu\left[2\alpha^2 h^{2k-1} \left|\left|w\right|\right|^2_{L^2(0,T;H^{k+1})} + h^{2k} \left|\left|w\right|\right|^2_{L^2(0,T;H^{k+1})}\right]\right. \\
&\left. + K\left[\alpha^2 h^{2k} |w(0)|^2_{k+1} + h^{2k+2} |w(0)|^2_{k+1} + 4\nu^{-1}\left(h^{2k+2} \left|\left|w_t\right|\right|^2_{L^2(0,T;H^{k+1})} + h^{2k} \left|\left|P\right|\right|^2_{L^2(0,T;H^k)}\right)\right]\right),
\end{aligned}
$$

and

$$
\begin{aligned}
&||e^u||^2 \\
\le & C\left(\alpha^2 h^{2k}|w(0)|_{k+1}^2 + h^{2k+2}|w(0)|_{k+1}^2 + \nu\left[2\alpha^2 h^{2k-1}\,||w||_{L^2(0,T;H^{k+1})}^2 + h^{2k}\,||w||_{L^2(0,T;H^{k+1})}^2\right]\right.\\
&+ K\left[\alpha^2 h^{2k}|w(0)|_{k+1}^2 + h^{2k+2}|w(0)|_{k+1}^2 + 4\nu^{-1}\left(h^{2k+2}\,||w_t||_{L^2(0,T;H^{k+1})}^2 + h^{2k}\,||P||_{L^2(0,T;H^k)}^2\right)\right]\\
&\left.+ 3h^{2k+2}|u|_{L^2(0,T;H^{k+1})}^2\right),
\end{aligned}
$$

where K is defined in Theorem 12.

Proof. The result follows by applying standard interpolation estimates to Theorem 3.4 with the selected discrete spaces. ■

Remark 14. *Theorem 12 provides an optimal $L^2(0,T;H^1)$ error estimate for w if $\alpha = O(h)$ (which we assume is the case). An additional feature is that the Gronwall constant does not depend explicitly on the inverse of the viscosity (which is not known to be true for discretizations of related models such as NS-α or Leray-α [63]).*

Proof. We will break up the proof for Theorem 12 into steps.

Step 1: Develop error equation.

We begin by testing (1.2) by $v_h \in V_h$ and (1.4) by $\chi_h \in V_h$ and subtracting them from (4.1)-(4.3) to get

$$
(u_t - (u_h)_t, v_h) + c(w,w,v_h) - c(w_h,w_h,v_h) - (P - q_h, \nabla\cdot v_h) + \nu\left(\nabla e^u, \nabla v_h\right) = 0, \tag{4.16}
$$

$$
\alpha^2\left(\nabla e^w, \nabla\chi_h\right) + (e^w, \chi_h) = (e^u, \chi_h), \tag{4.17}
$$

for any $v_h, \chi_h \in V_h$. Note that $(-\lambda_h, \nabla\cdot\chi_h) = 0$ since $\chi_h \in V_h$. We also obtain $(P - q_h, \nabla\cdot v_h)$ from

$$
(P - P_h, \nabla\cdot v_h) = (P - P_h, \nabla\cdot v_h) + (P_h - q_h, \nabla\cdot v_h) = (P - q_h, \nabla\cdot v_h),
$$

which holds for any $q_h \in Q_h$, since $v_h \in V_h$. Then after some manipulation (4.16) becomes

$$\begin{aligned}((\phi_h^u)_t, \phi_h^w) + \nu\,(\nabla\phi_h^u, \nabla\phi_h^w) &= -\,(\eta_t^u, \phi_h^w) - (c(w, w, \phi_h^w) \\ &\quad - c(w_h, w_h, \phi_h^w)) + (P - q_h, \nabla\cdot\phi_h^w)\,.\end{aligned} \tag{4.18}$$

Step 2: Finding suitable expressions for $(\nabla\phi_h^u, \nabla\phi_h^w)$ and $((\phi_h^u)_t, \phi_h^w)$ in equation (4.18).

First, we use the orthogonality of the Stokes operator (2.2) to get

$$(\nabla\phi_h^w, \nabla\eta^w) = \left(\nabla\phi_h^w, \nabla(w - I_h^{St} w)\right) = 0, \tag{4.19}$$

since $\phi_h^w \in V_h$. Next we test (4.17) with $\chi_h = A_h\phi_h^w$ and use (2.1) to get

$$\alpha^2\,(\nabla e^w, \nabla A_h\phi_h^w) + (e^w, A_h\phi_h^w) = (e^u, A_h\phi_h^w)\,.$$

We use $e^w = \phi_h^w + \eta^w$ to separate each term, then apply (4.19) to get

$$(\nabla\phi_h^u, \nabla\phi_h^w) = \alpha^2\,||A_h\phi_h^w||^2 + ||\nabla\phi_h^w||^2\,. \tag{4.20}$$

Now we must handle $((\phi_h^u)_t, \phi_h^w)$. We test (4.17) with $\chi_h = \phi_h^w$ and apply (2.1) to get

$$(\phi_h^u, \phi_h^w) + (\eta^u, \phi_h^w) = \alpha^2\,||\nabla\phi_h^w||^2 + ||\phi_h^w||^2 + (\eta^w, \phi_h^w)\,. \tag{4.21}$$

Next, take the derivative with respect to t of (4.21) and simplify,

$$\begin{aligned}((\phi_h^u)_t, \phi_h^w) &= \frac{d}{dt}[\alpha^2\,||\nabla\phi_h^w||^2 + ||\phi_h^w||^2 + (\eta^w, \phi_h^w) - (\eta^u, \phi_h^w)] - (\phi_h^u, (\phi_h^w)_t) \\ &= \frac{d}{dt}\left[\alpha^2\,||\nabla\phi_h^w||^2 + ||\phi_h^w||^2\right] + (\eta_t^w, \phi_h^w) + (\eta^w, (\phi_h^w)_t) \\ &\quad - (\eta_t^u, \phi_h^w) - (\eta^u, (\phi_h^w)_t) - (\phi_h^u, (\phi_h^w)_t)\,.\end{aligned} \tag{4.22}$$

Step 3: Finding a suitable expression for $(\phi_h^u, (\phi_h^w)_t)$ from (4.22).

Similar to what we did in (4.19), we will do the same for $((\phi_h^w)_t, \eta^w)$ and use (2.2) to get

$$(\nabla(\phi_h^w)_t, \nabla\eta^w) = \big(\nabla(\phi_h^w)_t, \nabla(w_t - I_h^{St}w)\big) = 0. \tag{4.23}$$

Now we test (4.17) with $\chi_h = (\phi_h^w)_t$ and use (4.23) to get

$$(\phi_h^u, (\phi_h^w)_t) = \frac{\alpha^2}{2}\frac{d}{dt}\,||\nabla\phi_h^w||^2 + \frac{1}{2}\frac{d}{dt}\,||\phi_h^w||^2 + (\eta^w, (\phi_h^w)_t) - (\eta^u, (\phi_h^w)_t)\,. \tag{4.24}$$

Substituting (4.24) into (4.22) and rearranging yields

$$((\phi_h^u)_t, \phi_h^w) = \frac{1}{2}\frac{d}{dt}\left[\alpha^2\,||\nabla\phi_h^w||^2 + ||\phi_h^w||^2\right] + (\eta_t^w, \phi_h^w) - (\eta_t^u, \phi_h^w)\,. \tag{4.25}$$

Now combine (4.25) and (4.20) to get the identity

$$\begin{aligned}&((\phi_h^u)_t, \phi_h^w) + \nu\,(\nabla\phi_h^u, \nabla\phi_h^w)\\ =&\frac{1}{2}\frac{d}{dt}\left[\alpha^2\,||\nabla\phi_h^w||^2 + ||\phi_h^w||^2\right] + (\eta_t^w, \phi_h^w) - (\eta_t^u, \phi_h^w) + \nu\left[\alpha^2\,||A_h\phi_h^w||^2 + ||\nabla\phi_h^w||^2\right].\end{aligned} \tag{4.26}$$

Rearranging (4.26) now provides

$$\begin{aligned}&\frac{1}{2}\frac{d}{dt}\left[\alpha^2\,||\nabla\phi_h^w||^2 + ||\phi_h^w||^2\right] + \nu\left[\alpha^2\,||A_h\phi_h^w||^2 + ||\nabla\phi_h^w||^2\right]\\ &= ((\phi_h^u)_t, \phi_h^w) + \nu\,(\nabla\phi_h^u, \nabla\phi_h^w) - (\eta_t^w, \phi_h^w) + (\eta_t^u, \phi_h^w)\,.\end{aligned} \tag{4.27}$$

Putting together (4.18) and (4.27) and canceling the (η_t^u, ϕ_h^w) term yields

$$\begin{aligned}&\frac{1}{2}\frac{d}{dt}\left[\alpha^2\,||\nabla\phi_h^w||^2 + ||\phi_h^w||^2\right] + \nu\left[\alpha^2\,||A_h\phi_h^w||^2 + ||\nabla\phi_h^w||^2\right]\\ &= -\,(c(w, w, \phi_h^w) - c(w_h, w_h, \phi_h^w)) + (P - q_h, \nabla\cdot\phi_h^w) - (\eta_t^w, \phi_h^w)\,.\end{aligned} \tag{4.28}$$

Step 4: Estimates for each term in equation (4.28).

Estimates for the right hand side of (4.28) where we employ the Cauchy-Schwarz inequality and Young's inequality are as follows:

$$|\,(\eta_t^w,\phi_h^w)\,| \leq 2\nu^{-1}\,||\eta_t^w||_{X'}^2 + \frac{\nu}{2}\,||\phi_h^w||^2\,, \tag{4.29}$$

$$|\,(P-q_h,\nabla\cdot\phi_h^w)\,| \leq 2\nu^{-1}\,||P-q_h||^2 + \frac{\nu}{2}\,||\nabla\phi_h^w||^2\,. \tag{4.30}$$

From Theorem 3.2 in [68], we have the following bound for the nonlinear terms,

$$\begin{aligned}
&|c(w,w,\phi_h^w) - c(w_h,w_h,\phi_h^w)| \\
&\qquad\leq C\left(\,||\eta^w||^2\,||w||_{L^\infty} + ||\nabla w||_{L^\infty}\,||\eta^w||^2 + [\,||w||_{L^\infty} + ||\nabla w||_{L^\infty}]\,||\phi_h^w||^2\right).
\end{aligned}$$

Applying (4.29)-(4.30) to (4.28) gives us

$$\begin{aligned}
&\frac{1}{2}\frac{d}{dt}\left[\alpha^2\,||\nabla\phi_h^w||^2 + ||\phi_h^w||^2\right] + \nu\left[\alpha^2\,||A_h\phi_h^w||^2 + ||\nabla\phi_h^w||^2\right] \\
\leq& C\left(\,||\eta^w||^2\,||w||_{L^\infty} + ||\nabla w||_{L^\infty}\,||\eta^w||^2 + [\,||w||_{L^\infty} + ||\nabla w||_{L^\infty}]\,||\phi_h^w||^2\right) \\
&+ 2\nu^{-1}\left(\,||\eta^w||_{X'}^2 + ||P-q_n||^2\right) + \frac{\nu}{2}\,||\phi_h^w||^2 + \frac{\nu}{2}\,||\nabla\phi_h^w||^2\,,
\end{aligned}$$

and then by rearranging we get

$$\begin{aligned}
&\frac{d}{dt}\left[\alpha^2\,||\nabla\phi_h^w||^2 + ||\phi_h^w||^2\right] + \nu\left[2\alpha^2\,||A_h\phi_h^w||^2 + ||\nabla\phi_h^w||^2\right] \\
\leq& 2C\left(\,||\eta^w||^2\,||w||_{L^\infty} + ||\nabla w||_{L^\infty}\,||\eta^w||^2\right) + 2\left(C\left[\,||w||_{L^\infty} + ||\nabla w||_{L^\infty}\right] + \nu\right)||\phi_h^w||^2 \\
&+ 4\nu^{-1}\left(\,||\eta_t^w||_{X'}^2 + ||P-q_n||^2\right).
\end{aligned}$$

Finally apply Gronwall's inequality and get

$$\begin{aligned}
&\alpha^2\,||\nabla\phi_h^w(T)||^2 + ||\phi_h^w(T)||^2 + \nu\int_0^T\left[2\alpha^2\,||A_h\phi_h^w||^2 + ||\nabla\phi_h^w||^2\right]dt \\
\leq& K\left[\alpha^2\,||\nabla\phi_h^w(0)||^2 + ||\phi_h^w(0)||^2 + 4\nu^{-1}\left(||\eta_t^w||_{L^2(0,T;X')}^2 + \inf_{q_h\in L^2(0,T;Q_h)}||P-q_h||_{L^2(0,T;L^2)}^2\right)\right],
\end{aligned}$$

where

$$K = \exp\left[2C\left(\,||w||_{L^1(0,T;L^\infty)} + ||\nabla w||_{L^1(0,T;L^\infty)}\right) + T\nu\right].$$

By the triangle inequality, we get the desired error term

$$
\begin{aligned}
&\alpha^2 \left|\left|\nabla e^w(T)\right|\right|^2 + \left|\left|e^w(T)\right|\right|^2 + \nu \int_0^T \left[2\alpha^2 \left|\left|A_h e^w\right|\right|^2 + \left|\left|\nabla e^w\right|\right|^2\right] dt \\
\leq &\alpha^2 \left|\left|\nabla \eta^w(0)\right|\right|^2 + \left|\left|\eta^w(0)\right|\right|^2 + \nu \left[2\alpha^2 \left|\left|A_h \eta^w\right|\right|^2_{L^2(0,T;L^2)} + \left|\left|\nabla \eta^w\right|\right|^2_{L^2(0,T;L^2)}\right] \\
&+ K \left[\alpha^2 \left|\left|\nabla \phi_h^w(0)\right|\right|^2 + \left|\left|\phi_h^w(0)\right|\right|^2 4\nu^{-1} \left(\left|\left|\eta_t^w\right|\right|^2_{L^2(0,T;X')} + \inf_{q_h \in L^2(0,T;Q_h)} \left|\left|P - q_h\right|\right|^2_{L^2(0,T;L^2)}\right)\right].
\end{aligned}
\tag{4.31}
$$

Step 5: Use the bound on e^w to find a bound for e^u.

Set $\chi_h = \phi_h^u$ in (4.3) and expand e^u to get

$$
\left|\left|\phi_h^u\right|\right|^2 = \alpha^2 \left(\nabla e^w, \nabla \phi_h^u\right) + \left(e^w, \phi_h^u\right) - \left(\eta^u, \phi_h^u\right).
$$

Applying the Cauchy-Schwarz inequality gives

$$
\left|\left|\phi_h^u\right|\right|^2 \leq \alpha^2 \left|\left|\nabla e^w\right|\right| \left|\left|\nabla \phi_h^u\right|\right| + \left|\left|e^w\right|\right| \left|\left|\phi_h^u\right|\right| + \left|\left|\eta^u\right|\right| \left|\left|\phi_h^u\right|\right|.
$$

Now use the inverse inequality on the first term of the left hand side,

$$
\left|\left|\phi_h^u\right|\right|^2 \leq C\alpha^2 \frac{1}{h} \left|\left|\nabla e^w\right|\right| \left|\left|\phi_h^u\right|\right| + \left|\left|e^w\right|\right| \left|\left|\phi_h^u\right|\right| + \left|\left|\eta^u\right|\right| \left|\left|\phi_h^u\right|\right|.
$$

Recall from Remark 3.1 that $\alpha = O(h)$, and then we reduce and evaluate at T to get

$$
\left|\left|\phi_h^u(T)\right|\right| \leq C \left(\alpha \left|\left|\nabla e^w(T)\right|\right| + \left|\left|e^w\right|\right|(T)\right) + \left|\left|\eta^w(T)\right|\right|.
$$

Squaring both sides and using Young's inequality on the right hand side gives us

$$
\left|\left|\phi_h^u(T)\right|\right|^2 \leq 2C \left(\alpha^2 \left|\left|\nabla e^w(T)\right|\right|^2 + \left|\left|e^w(T)\right|\right|^2\right) + 2 \left|\left|\eta^u(T)\right|\right|^2.
$$

For our error bound, we get

$$
\left|\left|e^u(T)\right|\right| \leq 2C \left(\alpha^2 \left|\left|\nabla e^w(T)\right|\right|^2 + \left|\left|e^w(T)\right|\right|^2\right) + 3 \left|\left|\eta^u(T)\right|\right|^2.
$$

Now applying (4.31) and (2.4) gives us the stated result.

■

4.2 Numerical experiments

In this section, we provide results for several numerical experiments that test EMAC-Reg and compare it to NSE schemes and other related models. Define α to be the filtering radius of the Helmholtz filter, typically chosen as $\alpha = O(h)$. The schemes that we utilize in addition to EMAC-Reg are as follows:

SKEW

Find $(u_h, p_h) \in (X_h, Q_h)$ such that for every $(v_h, q_h) \in (X_h, Q_h)$:

$$\begin{aligned} ((u_h)_t, v_h) + (u_h \cdot \nabla u_h, v_h) + \frac{1}{2}((\nabla \cdot u_h)u_h, v_h) - (p_h, \nabla \cdot v_h) + \nu (\nabla u_h, \nabla v_h) &= (f, v_h), \\ (\nabla \cdot u_h, q_h) &= 0. \end{aligned}$$

EMAC

Find $(u_h, p_h) \in (X_h, Q_h)$ such that for every $(v_h, q_h) \in (X_h, Q_h)$:

$$\begin{aligned} ((u_h)_t, v_h) + 2(D(u_h)u_h, v_h) + ((\nabla \cdot u_h)u_h, v_h) - (p_h, \nabla \cdot v_h) + \nu (\nabla u_h, \nabla v_h) &= (f, v_h), \\ (\nabla \cdot u_h, q_h) &= 0. \end{aligned}$$

NS-α

Find $(u_h, p_h, w_h, \lambda_h) \in (X_h, Q_h, X_h, Q_h)$ such that for every
$(v_h, q_h, \chi_h, r_h) \in (X_h, Q_h, X_h, Q_h)$:

$$\begin{aligned} ((u_h)_t, v_h) + (u_h \times w_h, v_h) - (p_h, \nabla \cdot v_h) + \nu (\nabla u_h, \nabla v_h) &= (f, v_h), \\ (\nabla \cdot u_h, q_h) &= 0, \\ (\lambda_h, \nabla \cdot \chi_h) + \alpha^2 (\nabla w_h, \nabla \chi_h) + (w_h, \chi_h) &= (u_h, \chi_h), \\ (\nabla \cdot w_h, r_h) &= 0. \end{aligned}$$

4.2.1 Convergence rate test for a problem with analytical solution

We now test the convergence results described by Theorem 12 on a problem that is analogous to the Chorin problem in [24], however it is slightly adjusted for the EMAC-Reg model. We deduce that for $\alpha \geq 0$, the filtered velocity, velocity and pressure defined by

$$\begin{aligned} w &= \begin{bmatrix} -\cos(\pi x)\sin(\pi y) \\ \sin(\pi x)\cos(\pi y) \end{bmatrix} e^{-2\pi^2 \nu t}, \\ u &= (1 + 2\pi^2\alpha^2)w, \\ p &= -w \cdot \nabla w, \end{aligned}$$

is a solution for (1.2)-(1.4) with $f = 0$ and initial conditions $w_0 = w(0)$, $u_0 = u(0)$. We consider the domain $\Omega = (0,1)^2$ and enforce the appropriate Dirichlet boundary conditions for u and w.

We will be conducting three experiments: a spatial convergence test with Δt fixed, a temporal convergence test with h fixed, and then a hybrid of the two. For each test, we take $\alpha = \frac{h}{2}$, $\nu = .2$ and use (P_2, P_1) elements on a uniform mesh. For the spatial convergence test, we use $\Delta t = .005$ with Crank-Nicolson time stepping and fix our end time at $T = 1$. We calculate the error with $h = \left(\frac{1}{2}\right)^i$ where $i = 1, \dots, 6$.

For the temporal convergence test, we have nearly identical parameters as the spatial convergence test. We set $h = \frac{1}{128}$ and $\Delta t = 2^{-i}$ for $i = 3, \dots, 6$.

For the hybrid experiment, we will use incrementing values for both h and Δt. We will start with $h = \frac{1}{4}$ and $\Delta t = \frac{1}{2}$, then $h = \frac{1}{8}$ and $\Delta t = \frac{1}{4}$, and so on. For each of these experiments, we will be calculating $L^\infty(0,T;L^2)$ and $L^\infty(0,T;H^1)$

We expect second order convergence for the H^1 norm, and an L^2 lift is expected to make the L^2 convergence third order for both tests. The results are displayed in Tables 4.1, 4.2, and 4.2 for spatial, temporal, and the combined convergence respectively. We observe third order accuracy in the L^2 norm and second order accuracy in the H^1 norm for spatial convergence, as expected. For temporal convergence, we observe second order convergence at the least for L^2 and mixed convergence rates for H^1. We get high convergence rates starting around $\Delta t = \frac{1}{4}$ through $\Delta t = \frac{1}{32}$, but both maintain at least second order convergence. We get similar results for the hybrid experiment.

h	$\|\|w - w_h\|\|_{\infty,0}$	Rate	$\|\|w - w_h\|\|_{\infty,1}$	Rate	$\|\|u - u_h\|\|_{\infty,0}$	Rate
1/2	8.98240e-04	-	1.33377e-02	-	8.98240e-04	-
1/4	1.07331e-04	3.06503	3.55447e-03	1.90780	1.07331e-03	3.06503
1/8	1.30963e-05	3.03484	9.11945e-04	1.96262	1.30963e-05	3.03484
1/16	1.62923e-06	3.00689	2.29787e-04	1.98865	1.62923e-06	3.00689
1/32	2.04701e-07	2.99260	5.75694e-05	1.99692	2.04701e-07	2.99260
1/64	3.32277e-08	2.62306	1.47046e-05	1.96904	3.32277e-08	2.62306

Table 4.1: Spatial convergence results for both u and w for EMAC-Reg

Δt	$\|\|w - w_h\|\|_{\infty,0}$	Rate	$\|\|w - w_h\|\|_{\infty,1}$	Rate	$\|\|u - u_h\|\|_{\infty,0}$	Rate
1/8	1.78485e-05	-	5.79085e-04	-	1.78485e-05	-
1/16	3.24554e-06	2.45927	6.32790e-05	3.19398	3.24554e-06	2.45927
1/32	8.03788e-07	2.01357	1.29501e-05	2.28876	8.03788e-07	2.01357
1/64	2.01230e-07	1.99797	7.41362e-06	0.80472	2.01230e-07	1.99797

Table 4.2: Temporal convergence results for both u and w for EMAC-Reg

h	Δt	$\|\|w - w_h\|\|_{\infty,0}$	Rate	$\|\|w - w_h\|\|_{\infty,1}$	Rate	$\|\|u - u_h\|\|_{\infty,0}$	Rate
1/4	1/2	6.37691e-03	-	0.11360e-00	-	6.37691e-03	-
1/8	1/4	4.71124e-04	3.75868	2.00927e-02	2.49922	4.71124e-04	3.75868
1/16	1/8	2.99146e-05	3.97719	2.89569e-03	2.79469	2.99146e-05	3.97719
1/32	1/16	3.59222e-06	3.05790	3.83564e-04	2.91637	3.59222e-06	3.05790
1/64	1/32	8.09962e-07	2.14895	5.08696e-05	2.91459	8.09962e-07	2.14895
1/128	1/64	2.01230e-07	2.00901	7.41362e-06	2.77855	2.01230e-07	2.00901

Table 4.3: Convergence results for decreasing values of h and Δt for u and w for EMAC-Reg

4.2.2 Gresho problem

Our next experiment is the Gresho Problem, which is also referred to as the standing vortex problem [39, 65, 87]. We start with an initial condition u_0 which is a solution of the steady Euler equations. Define

$r = \sqrt{x^2+y^2}$, and on $\Omega = (-0.5, 0.5)^2$, the velocity and pressure solutions are defined by

$$r \le .2 : \begin{cases} u = \begin{bmatrix} -5y \\ \\ 5x \end{bmatrix} \\ \\ p = 12.5r^2 + C_1 \end{cases}$$

$$.2 \le r \le .4 : \begin{cases} u = \begin{bmatrix} \frac{2y}{r} + 5y \\ \\ \frac{2x}{r} - 5x \end{bmatrix} \\ \\ p = 12.5r^2 - 20r + 4\log(r) + C_2 \end{cases}$$

$$r > .4 : \begin{cases} u = \begin{bmatrix} 0 \\ \\ 0 \end{bmatrix} \\ \\ p = 0 \end{cases}$$

where

$$\begin{aligned} C_2 &= (-12.5)(.4)^2 + 20(.4)^2 - 4\log(.4), \\ C_1 &= C_2 - 20(.2) + 4\log(.2). \end{aligned}$$

We compute solutions using EMAC-Reg, EMAC, NS-α, and SKEW with Crank-Nicolson time stepping and Newton iterations to solve the nonlinear problem, with $f = 0$ and no slip boundary conditions up to $T = 4.0$, and we take $\nu = 0$ (we also ran using $\nu = 10^{-7}$ and obtained the same results). We computed using (P_2, P_1) Taylor-Hood elements on a 48×48 uniform mesh, with a time step of $\Delta t = 0.01$. A spatial radius of $\alpha = \frac{1}{100}$ was used for EMAC-Reg and NS-α. Other values of α that were $O(h)$ were tested and gave similar results.

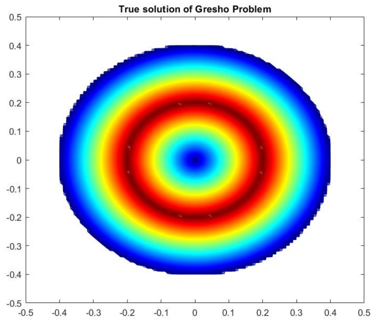

Figure 4.1: Speed contours of the true solution of the Gresho problem at all times.

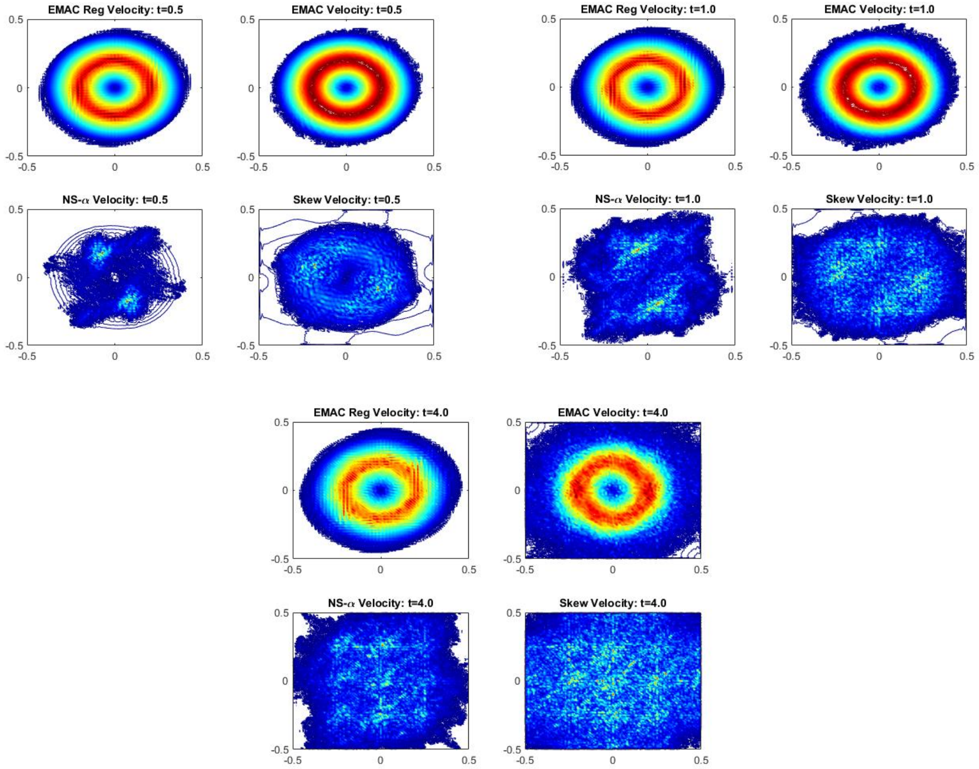

Figure 4.2: Velocity contours of our schemes at times $t = .5$, 1.0, and 4.0.

The Gresho solution is constant in time and so as time goes forward a good numerical solution will look similar to figure 4.1, which is the initial condition plot of speed.

At time $t = .5$, we observe from figure 4.2 that EMAC-Reg and EMAC perform equally well, whereas NS-α and SKEW are nearly unrecognizable compared to the true solution in figure 4.1. At time $t = 1$, we start to see EMAC-Reg outperform EMAC: EMAC-Reg approximately maintains the vortex shape, whereas the vortex for EMAC is spreading out with rough edges. At time $t = 4$, we see both vortices are spreading out, but it is clear that EMAC-Reg resembles figure 4.1 much better.

We also compare L^2 error and important physical quantities that we hope to be conserved: energy, momentum, and angular momentum versus time. We have shown in Theorem 11 that EMAC-Reg conserves energy, momentum, and angular momentum and it can be shown that SKEW and NS-α preserve energy [61, 63].

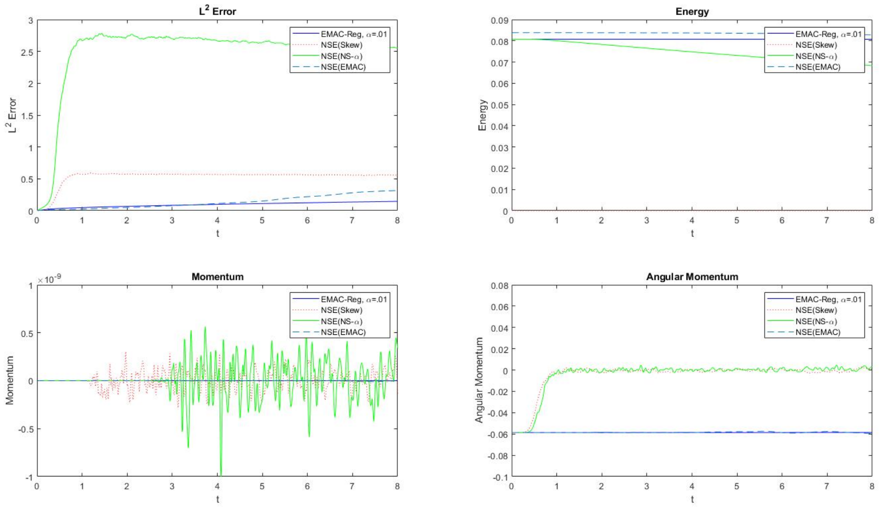

Figure 4.3: Plots of time versus L^2 error, energy, momentum and angular momentum

From figure 4.3 we observe that EMAC-Reg not only conserves energy, momentum, and angular momentum, but has significantly less L^2 error than the other formulations. We note also that EMAC was shown in [19] to have smaller L^2 error compared to several other formulations. We also observe that each of these schemes conserves energy, as expected. Momentum also appears to be conserved, noting the scale of the plot is $O(10^{-8})$. SKEW and NS-α appear to have large variation on a smaller scale, whereas EMAC and EMAC-Reg stay constant over time.

Then, as expected, SKEW and NS-α do not have constant angular momentum over time. EMAC approximately conserves angular momentum, and EMAC-Reg does the best job of conserving it.

4.2.3 2D Channel flow over a step

The next experiment is flow past a forward-backward facing step. We consider a 40 by 10 channel with a 1 by 1 step placed 5 units in. We enforce no-slip boundary conditions on the walls and step, and use a constant (with respect to time) parabolic inflow and outflow with peak velocity 1, viscosity $\nu = \frac{1}{600}$, and no external force, i.e., $f = 0$.

We ran this experiment using SKEW, EMAC, EMAC-Reg, and NS-α. We also ran SKEW on a fine mesh for reference, and note this solution matches the reference solution in [62]. For each model/scheme we used (P_2, P_1) elements, Crank-Nicolson time stepping, and Newton iterations to solve for the nonlinear term. We used a timestep of $\Delta t = .025$ and ran until the end time $T = 40$. A spatial filter of $\alpha = \frac{1}{10}$ was used for EMAC-Reg and NS-α. There are 3810 degrees of freedom on the coarse mesh and 145K degrees of freedom on the reference mesh, and hence the coarse mesh simulations represent significantly underresolved simulations.

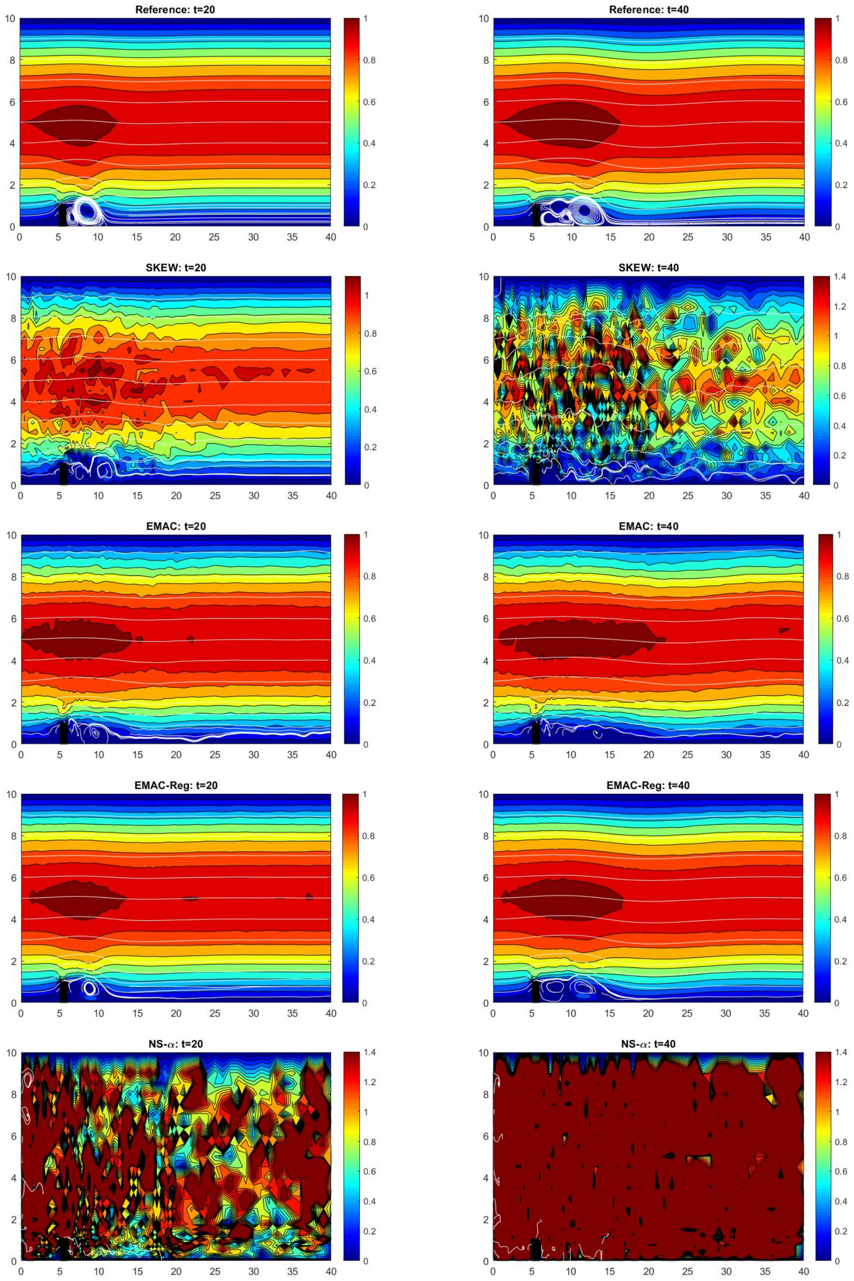

Figure 4.4: Shown above are speed contours of our 4 different schemes' solution plotted at times t=20 and t=40 with SKEW on a fine mesh as reference

SKEW provides an accurate and smooth solution on the fine mesh that matches results from [63] and we use as a reference solution. However it does not perform well on the coarse mesh.

EMAC provides a better solution than SKEW on the coarse mesh, but still exhibits oscillations and does not predict eddy detachment. EMAC-Reg provides the best coarse mesh solution, and qualitatively matches the reference solution quite well. The NS-α coarse mesh solution is very poor.

4.2.4 2D Kelvin-Helmholtz

Our final numerical test is a benchmark problem from [78] for 2D Kelvin-Helmholtz instability. Our domain is the unit square $\Omega = [0,1]^2$, with periodic boundary conditions along $x = 0$ and $x = 1$. The no penetration boundary condition $u \cdot n = 0$ is strongly enforced along $y = 0$ and $y = 1$. Weak enforcement of the free-slip condition is included as well. The initial condition is defined as

$$u_0(x,y) = \begin{bmatrix} u_\infty \tanh\left(2y - 1\delta_0\right) \\ 0 \end{bmatrix} + c_n \begin{bmatrix} \partial_y \phi(x,y) \\ -\partial_x \phi(x,y) \end{bmatrix},$$

where $\delta_0 = \frac{1}{28}$ is the initial vorticity thickness, the reference velocity $u_\infty = 1$, $c_n = 10^{-3}$ the noise/scaling factor, and

$$\phi(x,y) = u_\infty \exp\left(- \frac{(y-0.5)^2}{\delta_0^2} \right) \left(\cos(8\pi x) + \cos(20\pi x) \right).$$

The Reynolds number in this case is defined by $Re = \frac{\delta_o u_\infty}{\nu} = \frac{1}{28\nu}$.

We compute solutions for $Re = 1000$ using EMAC and EMAC-Reg; NS-α and SKEW simulations using similar parameter choices as EMAC-Reg became unstable, and we were unable to compute to the end-time, even trying several choices of α. It is possible that the better results would be found with stabilizations [25] or divergence free elements [13]. Taylor-Hood (P_2, P_1) elements are used for spatial discretization and BDF2 for temporal discretization with a uniform triangulation with $h = \frac{1}{48}$ for EMAC and EMAC-Reg, which gave us 19K velocity degrees of freedom. Note that we do lose some degree of property conservation by using BDF2 instead of Crank-Nicolson. We also computed a reference solution, which was computed using EMAC with $h = \frac{1}{128}$. For EMAC-Reg we computed using $\alpha = \frac{h}{3}$. We calculated up to $T = 10$ using step size $\Delta t = .001$. Newton's method was used to solve the nonlinear problems and were resolved typically in 2 to 3 steps.

Figure 4.5 displays enstrophy contours at different times. It was shown in [68] that SKEW does not

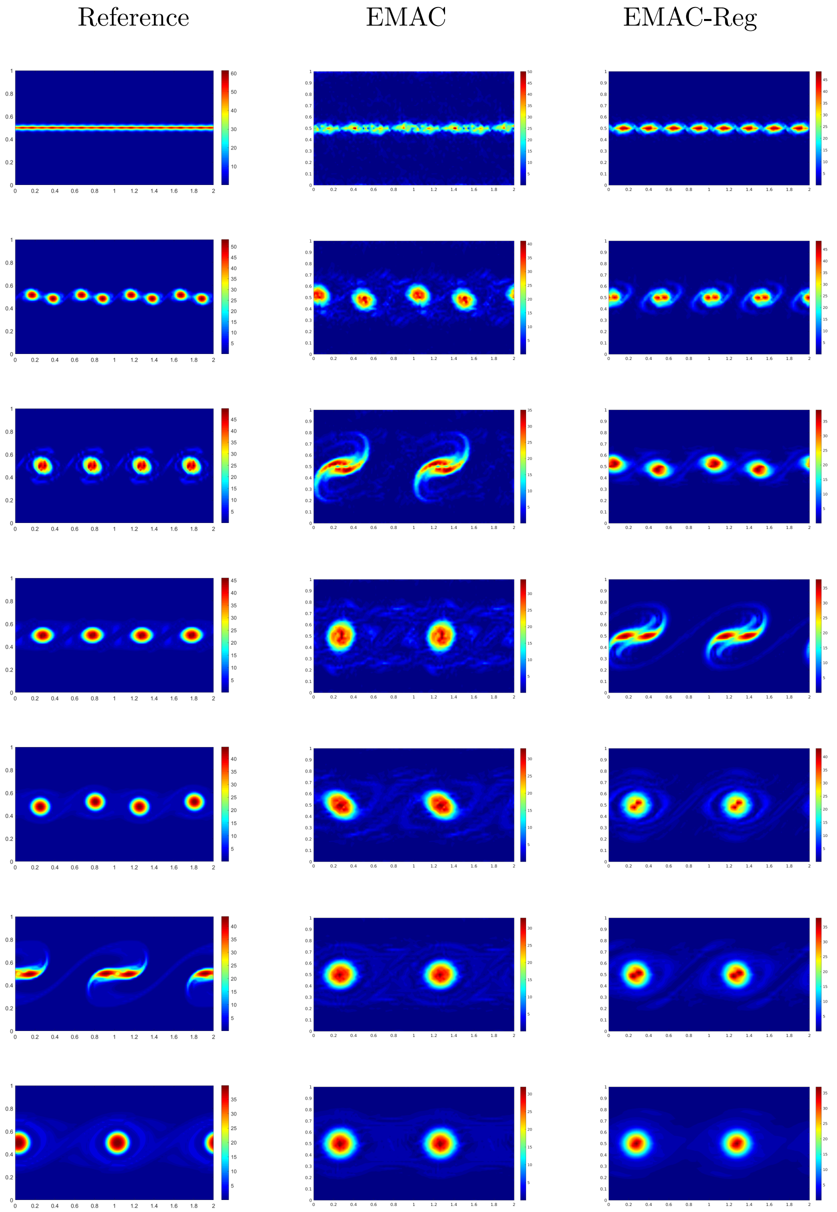

Figure 4.5: Shown above are the vorticity contours for the reference solution (left), EMAC (middle), and EMAC-Reg (right) with mesh size 256 for the reference and 48 for EMAC and EMAC-Reg, at times t =1, 2, 3, 4, 5, 6, 10, for Re=1000. Note the EMAC-Reg formulation used $\alpha = \frac{h}{3}$.

give resolved plots and show oscillations on a mesh significantly more refined than this one. We can expect that on a coarser mesh that SKEW performs even more poorly, which was made clear during the experiment and why we didn't include it in this proposal.

It was also shown in [68] that EMAC performed well on a finer mesh, however on the coarse mesh it does not predict important physical phenomena that happens earlier on in the simulation. Notice at time $t = 1$, EMAC-Reg shows clear vortices, whereas EMAC is more grainy. At times $t = 3$, 4 we see a more refined plot from EMAC-Reg of the vortices' behavior than EMAC.

Comparing to the reference solution, the earlier times for EMAC-Reg closely resemble the resolved solution, much more than EMAC. As mentioned above, we observe several small vortices in the early phases of the experiment for EMAC-Reg, which is precisely how it is depicted in [78]. We also see that EMAC-Reg converges to a steady solution closer to the time where the reference converges. We see convergence begin to form in the reference solution at $t = 6$, whereas EMAC converges around $t = 3$ and EMAC-Reg converges around $t = 4$.

Chapter 5

FOM-ROM inconsistency

5.1 Analysis of FOM-ROM inconsistency

In this chapter, we perform an error analysis that reveals how FOM-ROM inconsistency of the nonlinearity discretization creates additional error. To this end, we consider a ROM that is inconsistent with the FOM with respect to the nonlinearity discretization. Specifically, we assume that the FEM simulation was done with the SKEW form of the nonlinear term and EMAC is used in the ROM (in (2.21)). We note that the analysis that follows would hold in essentially the same way with any other combination of nonlinear forms that are not the same in the FEM and ROM schemes (although with additional difficulties if c is used).

Theorem 15. *Let $\{u_h^n\}_{n=1}^M$ be the FE solution from* (2.16)*, and $\{w_r^n\}_{n=1}^M$ be the ROM solution from* (2.21) *with EMAC nonlinear term and initial condition $w_r^0 = P_{L_2}^{V_r}(u_h^0)$, where $P_{L^2}^{V_r}$ is the L^2 projection onto the ROM space. Assuming that the FE solution satisfies $C_u = \max\{\|\nabla u_h^n\|\}_{n=1}^M$ is bounded independent of any discretization parameters and Δt is sufficiently small, we have the following error bound for the ROM as an approximation of the FOM:*

$$
\begin{aligned}
&\left|\left|w_r^M - u_h^M\right|\right|^2 + \nu\,\Delta t \sum_{n=1}^{M} ||\nabla(w_r^n - u_h^n)||^2 \\
&\leq Ce^{\left((\nu^{-3}C_u^4+\nu^{-1})T\right)}\left(\nu^{-1}(C_u^2+1)\sum_{k=r+1}^{d}\|\nabla\psi_k\|^2\lambda_k + \underbrace{C_u^2\left|\left|\left|\nabla\cdot\left(P_{L^2}^{V_r}(u_h^{n_c}) - w_r^{n_c}\right)\right|\right|\right|_{1,0} + C_u^2\,|||\nabla\cdot u_h^{n_c}|||_{2,0}^2}_{\text{arises from FOM-ROM inconsistency}}\right),
\end{aligned} \tag{5.1}
$$

where λ_k are the eigenvalues in the eigenvalue problem used to construct the ROM basis [60, 82], $T = M\Delta t$ is the final time, and n_c is the value for n which corresponds to $\max\{\|\nabla u_h^n\|\}_{n=1}^M$.

Proof of Theorem 15. Denote $e^n = u_h^n - w_r^n = u_h^n - P_{L^2}^{V_r}(u_h^n) + P_{L^2}^{V_r}(u_h^n) - w_r^n =: \eta^n + \phi_r^n$. Subtracting (2.21) from (2.16) with $v_h = v_r \in V_r$ yields

$$\frac{1}{\Delta t}(e_r^{n+1} - e_r^n, v_r) + b^*\left(u_h^{n+1}, u_h^{n+1}, v_r\right) - c(w_r^{n+1}, w_r^{n+1}, v_r) + \nu(\nabla e^{n+1}, \nabla v_r) = 0 \quad \forall v_r \in V_r, \tag{5.2}$$

with the pressure term vanishing thanks to $v_r \in V_r$ being discretely divergence free. Choosing $v_r = \phi_r^{n+1}$, we obtain

$$\begin{aligned} &\frac{1}{2\Delta t}\left(\left|\left|\phi_r^{n+1}\right|\right|^2 - ||\phi_r^n||^2\right) + \nu\left|\left|\nabla\phi_r^{n+1}\right|\right|^2 \leq \\ &-\left(b^*\left(u_h^{n+1}, u_h^{n+1}, \phi_r^{n+1}\right) - c(w_r^{n+1}, w_r^{n+1}, \phi_r^{n+1})\right) - \nu(\nabla\eta^{n+1}, \nabla\phi_r^{n+1}), \end{aligned} \tag{5.3}$$

using that $(\eta^{n+1} - \eta^n, \phi_r^{n+1}) = 0$ since η^n, η^{n+1} are L^2-orthogonal to ϕ_r^{n+1} by construction. The last term on the right-hand side of (5.3) is bounded in the usual way with Cauchy-Schwarz and Young's inequalities via

$$\begin{aligned} -\nu(\nabla\eta^{n+1}, \nabla\phi_r^{n+1}) &\leq \nu\|\nabla\eta^{n+1}\|\|\nabla\phi_r^{n+1}\| \\ &\leq 2\nu\|\nabla\eta^{n+1}\|^2 + \frac{\nu}{8}\|\nabla\phi_r^{n+1}\|^2. \end{aligned}$$

We begin the analysis of the nonlinear terms by adding and subtracting $c(u_h^{n+1}, u_h^{n+1}, \phi_r^{n+1})$ and then applying the triangle inequality to get

$$\begin{aligned} &|b^*\left(u_h^{n+1}, u_h^{n+1}, \phi_r^{n+1}\right) - c(w_r^{n+1}, w_r^{n+1}, \phi_r^{n+1})| \\ &\quad \leq |b^*(u_h^{n+1}, u_h^{n+1}, \phi_r^{n+1}) - c(u_h^{n+1}, u_h^{n+1}, \phi_r^{n+1})| + |c(u_h^{n+1}, u_h^{n+1}, \phi_r^{n+1}) - c(w_r^{n+1}, w_r^{n+1}, \phi_r^{n+1})|. \end{aligned} \tag{5.4}$$

For the second term on the right-hand side of (5.4), we follow analysis from [68] to obtain

$$
\begin{aligned}
&|c(u_h^{n+1}, u_h^{n+1}, \phi_r^{n+1}) - c(w_r^{n+1}, w_r^{n+1}, \phi_r^{n+1})| \\
&\qquad \leq \; C\nu^{-1} \left[\left|\left|\nabla u_h^{n+1}\right|\right|^2 \left|\left|\eta^{n+1}\right|\right| \left|\left|\nabla \eta^{n+1}\right|\right| + \left|\left|u_h^{n+1}\right|\right| \left|\left|\nabla u_h^{n+1}\right|\right| \left|\left|\nabla \eta^{n+1}\right|\right|^2 \right] \\
&\qquad\qquad + M_0 \nu^{-3} \left|\left|\nabla u_h^{n+1}\right|\right|^4 \left|\left|\phi_r^{n+1}\right|\right|^2 + \frac{\nu}{8} \left|\left|\nabla \phi_r^{n+1}\right|\right|^2, \quad (5.5)
\end{aligned}
$$

where M_0 is a constant depending only on the size of Ω.

To bound the first term on the right-hand side of (5.4), we expand the definitions of b^* and c and use that

$$
\begin{aligned}
2(D(u_h^{n+1})u_h^{n+1}, \phi_r^{n+1}) = ((\nabla u_h^{n+1})u_h^{n+1}, \phi_r^{n+1}) + ((\nabla u_h^{n+1})^T u_h^{n+1}, \phi_r^{n+1}) = \\
(u_h^{n+1} \cdot \nabla u_h^{n+1}, \phi_r^{n+1}) + (\phi_r^{n+1} \cdot \nabla u_h^{n+1}, u_h^{n+1})
\end{aligned}
$$

to get

$$
\begin{aligned}
&|b^* \left(u_h^{n+1}, u_h^{n+1}, \phi_r^{n+1}\right) - c(u_h^{n+1}, u_h^{n+1}, \phi_r^{n+1})| \\
=&|(u_h^{n+1} \cdot \nabla u_h^{n+1}, \phi_r^{n+1}) + \frac{1}{2}((\nabla \cdot u_h^{n+1})u_h^{n+1}, \phi_r^{n+1}) - (u_h^{n+1} \cdot \nabla u_h^{n+1}, \phi_r^{n+1}) \\
&\quad - (\phi_r^{n+1} \cdot \nabla u_h^{n+1}, u_h^{n+1}) - ((\nabla \cdot u_h^{n+1})u_h^{n+1}, \phi_r^{n+1})| \\
=&|(\phi_r^{n+1} \cdot \nabla u_h^{n+1}, u_h^{n+1}) + \frac{1}{2}((\nabla \cdot u_h^{n+1})u_h^{n+1}, \phi_r^{n+1})|. \qquad (5.6)
\end{aligned}
$$

Note that these terms arise only because of the inconsistency between the FOM and ROM, and we now bound both of them. For the first term, after using the divergence theorem [61] and then Hölder and Sobolev inequalities, we obtain

$$
\begin{aligned}
(\phi_r^{n+1} \cdot \nabla u_h^{n+1}, u_h^{n+1}) &= -\frac{1}{2}((\nabla \cdot \phi_r^{n+1})u_h^{n+1}, u_h^{n+1}) \\
&\leq \frac{1}{2}\|\nabla \cdot \phi_r^{n+1}\| \|u_h^{n+1}\|_{L^6} \|u_h^{n+1}\|_{L^3} \\
&\leq \frac{1}{2}M_0 \left|\left|\nabla \cdot \phi_r^{n+1}\right|\right| \left|\left|\nabla u_h^{n+1}\right|\right|^{\frac{3}{2}} \left|\left|u_h^{n+1}\right|\right|^{\frac{1}{2}}. \qquad (5.7)
\end{aligned}
$$

For the second term from (5.6), we denote C to be a generic constant depending only on the domain size, and apply Hölder's inequality, Sobolev inequalities, and Young's inequality to get

$$\begin{aligned}\frac{1}{2}((\nabla\cdot u_h^{n+1})u_h^{n+1},\phi_r^{n+1}) &\le \frac{1}{2}\left|\left|\nabla\cdot u_h^{n+1}\right|\right|\left|\left|u_h^{n+1}\right|\right|_{L^6}\left|\left|\phi_r^{n+1}\right|\right|_{L^3}\\ &\le C^{\frac{1}{2}}\left|\left|\nabla\cdot u_h^{n+1}\right|\right|\left|\left|\nabla u_h^{n+1}\right|\right|\left|\left|\phi_r^{n+1}\right|\right|^{1/2}\left|\left|\nabla\phi_r^{n+1}\right|\right|^{1/2}\\ &\le CC_u^2\left|\left|\nabla\cdot u_h^{n+1}\right|\right|^2+\left|\left|\phi_r^{n+1}\right|\right|\left|\left|\nabla\phi_r^{n+1}\right|\right|\\ &\le CC_u^2\left|\left|\nabla\cdot u_h^{n+1}\right|\right|^2+2\nu^{-1}\left|\left|\phi_r^{n+1}\right|\right|^2+\frac{\nu}{8}\left|\left|\nabla\phi_r^{n+1}\right|\right|^2. \qquad (5.8)\end{aligned}$$

Combining the above bounds and applying the Poincaré inequality, we get

$$\begin{aligned}\frac{1}{2\Delta t}\left(\left|\left|\phi_r^{n+1}\right|\right|^2-\left|\left|\phi_r^{n}\right|\right|^2\right)+\frac{\nu}{2}\left|\left|\nabla\phi_r^{n+1}\right|\right|^2 \le M_0\nu^{-3}\left|\left|\nabla u_h^{n+1}\right|\right|^4\left|\left|\phi_r^{n+1}\right|\right|^2+2\nu^{-1}\left|\left|\phi_r^{n+1}\right|\right|\\ +C\nu^{-1}\left(\left|\left|\nabla u_h^{n+1}\right|\right|^2\left|\left|\nabla\eta^{n+1}\right|\right|^2\right)+2\nu\left|\left|\nabla\eta^{n+1}\right|\right|^2\\ \underbrace{+\frac{1}{2}M_0\left|\left|\nabla\cdot\phi_r^{n+1}\right|\right|\left|\left|\nabla u_h^{n+1}\right|\right|^{\frac{3}{2}}\left|\left|u_h^{n+1}\right|\right|^{\frac{1}{2}}+CC_u^2\left|\left|\nabla\cdot u_h^{n+1}\right|\right|^2}_{\text{FOM-ROM inconsistency}}.\end{aligned}$$

Using the smoothness assumption on u_h and summing over time steps provides us with

$$\begin{aligned}\|\phi_r^M\|^2+\nu\Delta t\sum_{n=1}^{M}\left|\left|\nabla\phi_r^n\right|\right|^2 \le \|\phi_r^0\|^2+\Delta t\sum_{n=1}^{M}C(C_u^4\nu^{-3}+\nu^{-1})\left|\left|\phi_r^n\right|\right|^2+4\nu\Delta t\sum_{n=1}^{M}\left|\left|\nabla\eta^n\right|\right|^2\\ +C\nu^{-1}C_u^2\Delta t\sum_{n=1}^{M}\|\nabla\eta^n\|^2+\underbrace{CC_u^2\Delta t\sum_{n=1}^{M}\left(\left|\left|\nabla\cdot\phi_r^n\right|\right|+\left|\left|\nabla\cdot u_h^n\right|\right|^2\right)}_{\text{FOM-ROM inconsistency}}. \qquad (5.9)\end{aligned}$$

Note that $\phi_r^0=0$ by definition of the initial condition for the ROM being $w_r^0=P_{L_2}^{V_r}(u_h^0)$. Hence, we can now apply the discrete Gronwall inequality in Lemma 6, the POD projection error formula (2.22) [82], and the triangle inequality to find that, for sufficiently small Δt, we get a bound for the ROM as an approximation of the FOM:

$$\begin{aligned}&\left|\left|w_r^M-u_h^M\right|\right|^2+\nu\Delta t\sum_{n=1}^{M}\left|\left|\nabla(w_r^n-u_h^n)\right|\right|^2\\ &\quad\le Ce^{\left((\nu^{-3}C_u^4+\nu^{-1})T\right)}\left(\nu^{-1}(C_u^2+1)\sum_{k=r+1}^{d}\|\nabla\varphi_k\|^2\lambda_k+\underbrace{C_u^2\left|\left|\left|\nabla\cdot\phi_r^{n_c}\right|\right|\right|_{1,0}+C_u^2\left|\left|\left|\nabla\cdot u_h^{n_c}\right|\right|\right|_{2,0}^2}_{\text{FOM-ROM inconsistency}}\right), \qquad (5.10)\end{aligned}$$

where λ_k are the eigenvalues in the eigenvalue problem used to construct the ROM basis [60]. This completes the proof.

■

Remark 16. *For the smoothness assumption on u_h in the theorem, if for example (P_2, P_1) Taylor-Hood elements are used for the FE spatial discretization and a BDF2 scheme with $\Delta t \leq O(h^{1/2})$ is used for the time discretization, then using the inverse inequality, standard interpolation estimates, convergence of the FE* (2.20)*, and the H^1 stability of the L^2 projection into the FE space [12], $P_{L^2}^{X_h}$, we obtain*

$$\begin{aligned}
\|\nabla u_h^n\| &\leq \|\nabla(u_h^n - u^n)\| + \|\nabla u^n\| \\
&\leq \|\nabla(u_h^n - P_{L^2}^{X_h}(u^n))\| + \|\nabla(u^n - P_{L^2}^{X_h}(u^n))\| + \|\nabla u^n\| \\
&\leq Ch^{-1}\|u_h^n - P_{L^2}^{X_h}(u^n)\| + (1 + C_\Omega)\|\nabla u^n\| + \|\nabla u^n\| \\
&\leq Ch^{-1}\left(\|u_h^n - u^n\| + \|u^n - P_{L^2}^{X_h}(u^n)\|\right) + (2 + C_\Omega)\|\nabla u^n\| \\
&\leq Ch^{-1}\left(h^2 + \Delta t^2\right) + (2 + C_\Omega)\|\nabla u^n\| \\
&\leq C,
\end{aligned}$$

where C depends on the true solution and problem data but is independent of discretization parameters.

Remark 17. *The theorem reveals how the FOM-ROM inconsistency plays a role in the error: if there is FOM-ROM consistency, the last two terms on the right hand side of* (5.1) *vanish, and thus as r is increased the ROM solution recovers the FOM solution. However, if there is FOM-ROM inconsistency in the nonlinear formulation, then no matter how many modes are used, the ROM error will not be better than the divergence error from the FOM (which is generally not very small [55]). Thus, ROMs that are FOM inconsistent do not converge with respect to the ROM dimension, r.*

Remark 18. *The theorem shows that if the FOM gives divergence-free solutions (e.g., using Scott-Vogelius finite elements or others, see e.g. [55]), then the FOM-ROM inconsistency error is eliminated. This is perhaps not surprising, since with divergence-free elements all of the nonlinearity formulations are equivalent, up to a potential term. We note, however, that implementing divergence-free FEs can be challenging, and so the FOM-ROM inconsistency is relevant for the majority of FEs used in practical computations. We also note that, if grad-div stabilization is used in the FEM, then the FOM-ROM inconsistency error can be reduced since the divergence error of the FEM solution is reduced.*

5.2 Numerical Tests

We give results using channel flow around a cylinder. This experiment agrees with the above analysis: better results are obtained when the ROM formulation matches the FEM formulation whose snapshots created the ROM space, i.e., the FOM-ROM consistency of the nonlinearity discretization is beneficial.

The velocity-pressure spaces for the FOM (FEM simulations) are chosen to be (P_2, P_1) Taylor-Hood elements, along with BDF2 time stepping. The nonlinearity is resolved with Newton's method at each time step, in the FOMs and the ROMs. The FOM simulations create the snapshots, and we follow the procedure in [16] to construct the ROM modes and ROM space. We assemble the ROM operator corresponding to the nonlinearity in the NSE without using hyperreduction [8, 96]. We note, however, that hyperreduction may be needed for other types of PDEs.

In this section, we denote FOMs by EMAC-FEM, SKEW-FEM and CONV-FEM, and ROMs by EMAC-ROM, SKEW-ROM, and CONV-ROM.

5.2.1 Channel flow around a cylinder

We consider ROM simulations for 2D channel flow past a cylinder at Re=200. We follow the setup from [54, 77], and use a 2.2×0.41 rectangular channel domain containing a cylinder (circle) of radius 0.05 centered at $(0.2, 0.2)$; the domain is shown in Figure 5.1 and the Delaunay generated 2886 element triangular mesh is shown in Figure 5.2. There is no external forcing ($f = 0$), the kinematic viscosity is taken to be $\nu = 0.0005$, no-slip boundary conditions are prescribed for the walls and the cylinder, and the inflow and outflow velocity profiles are given by

$$u_1(0, y, t) = u_1(2.2, y, t) = \frac{6}{0.41^2} y(0.41 - y), \quad u_2(0, y, t) = u_2(2.2, y, t) = 0.$$

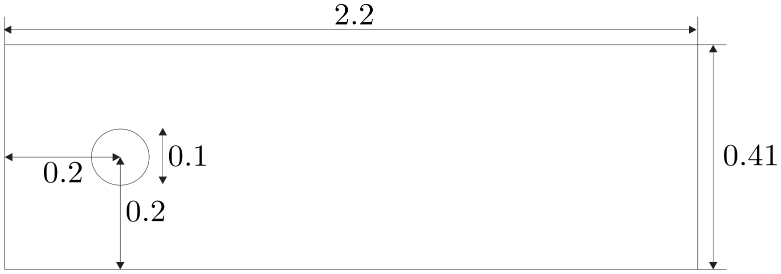

Figure 5.1: The domain for the channel flow past a cylinder numerical experiment.

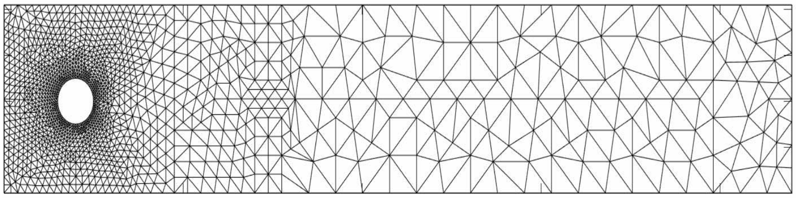

Figure 5.2: The mesh used for the FEM computations for channel flow past a cylinder numerical experiment.

Snapshots to create the ROMs are generated by computing with EMAC-FEM and CONV-FEM as the FOMs, both using (P_2, P_1) Taylor-Hood elements and BDF2 time stepping with $\Delta t = 0.001$. We use a relatively coarse mesh having 12K velocity degrees of freedom (dof), noting that, for many practical applications (e.g., engineering and geophysical turbulent flows), ROMs are created from FOMs that are not fully resolved [2]. Each FOM is started from rest, and run to $t = 15$. By $t = 4$, a periodic-in-time/statistically steady state is reached. ROM spaces are created from a full period of snapshots. Each ROM is run starting from the FOM solution at $t = 5$ (projected into the ROM space), and run for 10 time units (which is $t = 15$ for the FOM) using the same time step size as the FOM, $\Delta t = 0.001$. Comparisons of the ROMs and FOMs are done using energy and drag prediction, with energy at time t defined as $\frac{1}{2}\|u(t)\|^2$ and the drag coefficient defined by [54]

$$c_d(t) = 20 \int_S \left(\nu \frac{\partial u_{t_S}(t)}{\partial n} n_y - p(t) n_x \right) dS,$$

where S is the cylinder, $p(t)$ is pressure, u_{t_S} is the tangential velocity, the length scale is selected to be the diameter of the cylinder, and $n = \langle n_x, n_y \rangle$ is the outward unit normal.

We run multiple experiments with this test problem. In section 5.2.1.1, we use EMAC-FEM as the FOM and then run EMAC-ROM, SKEW-ROM and CONV-ROM and compare results (EMAC-ROM is superior). In section 5.2.1.2, we repeat this test but using CONV-FEM as the FOM (CONV-ROM). Finally, in section 5.2.1.3, we study convergence of the FOM to the ROM, as the number of modes N increases. We show that for EMAC-FEM as the FOM, EMAC-ROM converges to the FOM as N increases, but SKEW-ROM does not converge for increasing N and appears to lock (as predicted by our theory above).

5.2.1.1 Comparison of ROMs using EMAC-FEM as FOM

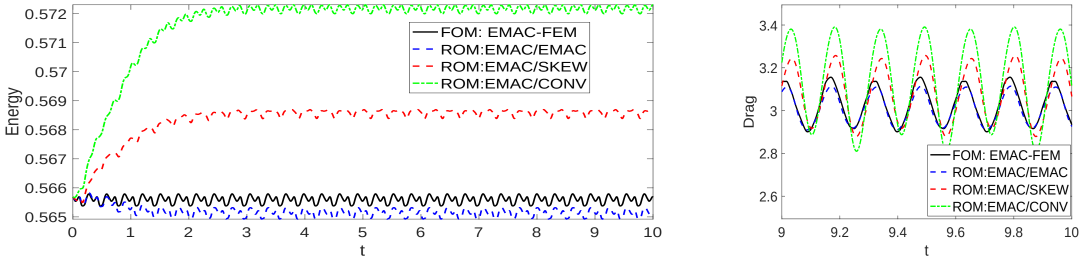

Figure 5.3: Shown above are results of ROM simulations built from EMAC-FEM as the FOM, using EMAC-ROM, SKEW-ROM, and CONV-ROM with 13 modes.

For our first test, we choose EMAC-FEM as the FOM to build the ROM space, and then run EMAC-ROM, SKEW-ROM and CONV-ROM, each with $N = 13$ modes. Results of each run's energy versus time and drag versus time are shown in Figure 5.3 (drag is shown only on $9 \leq t \leq 10$ so it is easier to see and compare, but differences are similar on $0 \leq t \leq 9$). From the plots, it is clear the EMAC-ROM (labeled ROM: EMAC/EMAC) is the most accurate for both energy and drag prediction, in terms of best matching the FOM energy and drag predictions. Thus, this is an excellent illustration of the theory above that shows that ROMs that are consistent with FOMs give better results.

5.2.1.2 Comparison of ROMs using CONV-FEM as FOM

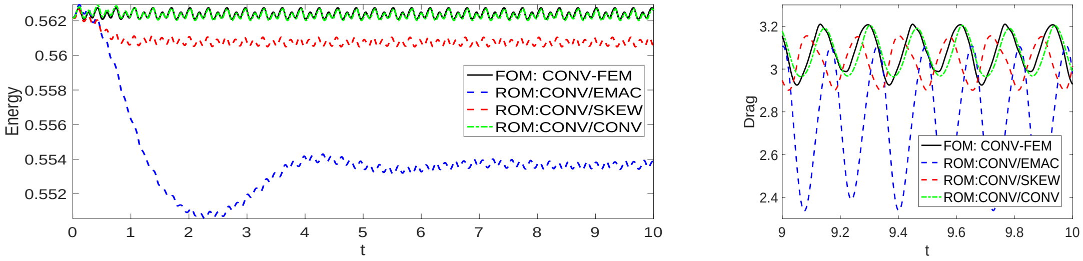

Figure 5.4: Shown above are results of ROM simulations built from CONV-FEM as the FOM, using EMAC-ROM, SKEW-ROM, and CONV-ROM with 13 modes.

For our second test, we choose CONV-FEM as the FOM to build the ROM space, and then run EMAC-ROM, SKEW-ROM and CONV-ROM, each with $N = 13$ modes. Results of each run's energy versus time and drag versus time are shown in Figure 5.4. From the plots, it is clear the CONV-ROM is the most accurate for both energy and drag prediction, in terms of best matching the FOM energy and drag predictions. This is another illustration of the theory above that shows that ROMs that are consistent with FOMs give better results.

5.2.1.3 Convergence of ROM to FOM depends on FOM-ROM consistency

Theorem 15 gives a bound that guarantees that a *consistent* ROM will converge to the FOM as the number of modes N increases. The bound also suggests that an inconsistent ROM may only converge to the FOM up to the terms depending on the divergence error in the FOM. This is analogous to locking phenomena in convergence theory for numerical methods, where as discretization parameters get small (and number of degrees of freedom gets large), the error cannot decrease below a certain level. We now illustrate this theory.

First, we consider the consistent case, and choose EMAC-FEM for the FOM, and EMAC-ROM as the ROM, and run the ROM using N=9, 13, and 16 modes. From the plots in Figure 5.5 at the top row, we observe convergence of the energy and drag to the FOM as N increases.

For the inconsistent case, we repeat the same test, again choosing EMAC-FEM as the FOM, but now choosing SKEW-ROM for the ROMs. The plots in Figure 5.5 at the bottom row do not show convergence to the FOM, but seem to show convergence to a different limit (since N=13 and 16 plots are nearly identical). This is in agreement with the theory that FOM-ROM inconsistency can prevent convergence of the ROM to the FOM as N increases.

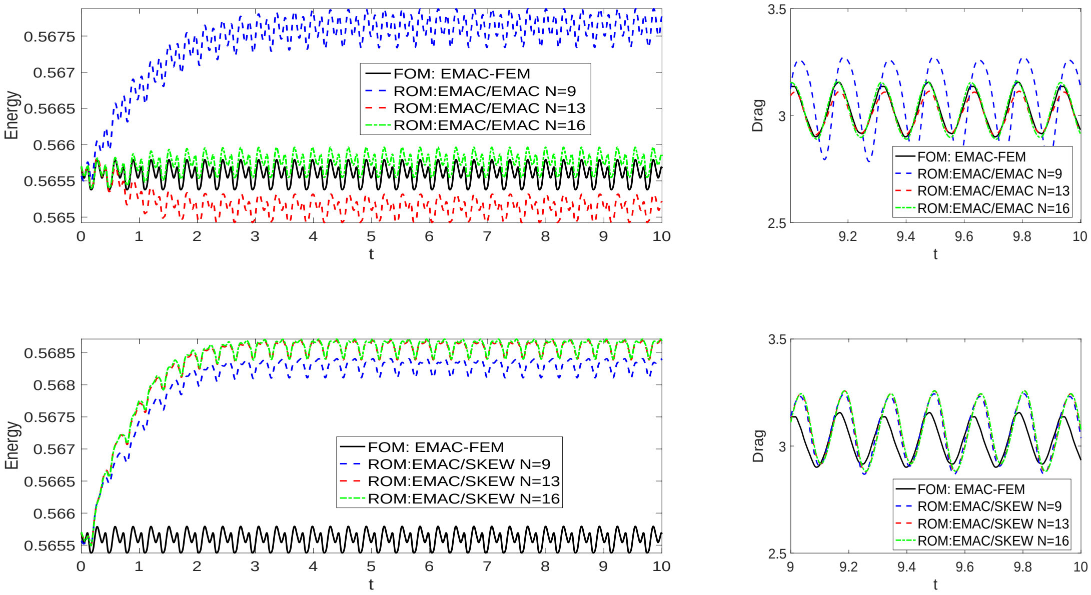

Figure 5.5: Shown above are results of ROM simulations built from EMAC-FEM as the FOM, and (top) EMAC-ROM and (bottom) SKEW-ROM with N=9, 13, and 16 modes.

Chapter 6

Anderson Acceleration in deal.II

In this chapter, we will implement methods of AA in the finite element library deal.II and compare it against Newton and Picard iterations using iteration count and computation time. What motivates this section is that we must perform a nonlinear solve at each timestep for EMAC (among other formulations) so it is critical to have an efficient nonlinear solver.

A standard method for solving the nonlinear problem is a Newton iteration. This is an obvious choice because it has quadratic convergence and is easy to implement if we can calculate derivatives. However, for larger Re, it is more challenging to find an appropriate initial guess. What many programs do to find a nice initial guess is a continuation method [94]. If $Re > 1000$ for the deal.II example tutorial step-57, the steady Navier-Stokes equations (SNSE) is solved using Newton with line search with $Re = 1000$. The solution is used as an initial guess for the same problem with $Re = 3000$. Re is increased by an increment of 2000 each iteration until the target Re is reached.

As one can imagine, this method will take a lot of time for large Re, so we would like to implement something better. Picard iterations with AA is precisely what we need. Picard iterations do not require a close initial guess, so we will not need to iterate through values of Re. Although, because of the linear convergence, we can expect low Re simulations to be slower with this method. However we will see that this method is preferable for high Re. Appendix B contains a link to the author's github profile which contains the programs used for the experiments in this chapter.

6.1 Step-57 Anderson acceleration implementation

6.1.1 Picard implementation

In this section we will discuss our implementation of AA in deal.II. We adjusted the step-57 tutorial program to run Picard iteration with Anderson acceleration steps (instead of Newton iterations). We begin by defining the steady Navier Stokes Equations:

$$\begin{aligned} -\nu\Delta u + (u\cdot\nabla)u + \nabla p &= f, \\ -\nabla\cdot u &= 0, \end{aligned} \tag{6.1}$$

where the variables and constants are defined the same as (1.1).

We also define the Picard iteration. Suppose we wish to solve the nonlinear problem

$$L(w) = f,$$

where L is some nonlinear differential operator and w is the solution. We rewrite this $L(w)$ such that

$$L(w) = G(w)w,$$

where G is some differential operator. Then, we are left with the algorithm:

Algorithm 19 (Picard iteration)**.** *For some differential operator G,*

```
Step 1: Choose $w_0$ (usually can be 0).
Step 2: For $k = 1, 2, \dots$
        Solve $G(w_k)w_{k+1} = f$.
```

In general with nonlinear solvers, we solve for w_{k+1} using the following equation:

$$w_{k+1} = w_k + \delta w_k,$$

where δw_k is our "update" and added to the previous iteration. Substituting w_{k+1} into Algorithm 19, we get

$$
\begin{aligned}
&G(w_k)(w_k + \delta w_k) = f \\
\Rightarrow &G(w_k)w_k + G(w_k)\delta w_k = f \\
\Rightarrow &G(w_k)\delta w_k = f - G(w_k)w_k. \qquad (6.2)
\end{aligned}
$$

Now equation 1 of (6.1) to the left hand side of (6.2),

$$G(u_k)\delta u_k = -\nu\Delta\delta u_k + u_k \cdot \nabla\delta u_k + \nabla\delta p_k,$$

where u_k and p_k are the kth iteration of u and p. This gives our final Picard iteration system

$$
\begin{aligned}
-\nu\Delta\delta u_k + u_k \cdot \nabla\delta u_k + \nabla\delta p_k &= f - G(u_k)u_k, \\
-\nabla \cdot \delta u_k &= \nabla \cdot u_k.
\end{aligned}
$$

6.1.2 Anderson acceleration implementation

Now that we have established the Picard implementation, recall Algorithm 4. The first step is to perform the standard fixed point method, then the goal for the second step is to minimize

$$\min_{\sum_{j=k-m}^{k} \alpha_j^{k+1}=1} \left\| \sum_{j=k-m}^{k} \alpha_j^{k+1}(\tilde{\mathbf{u}}_{j+1} - \mathbf{u}_j) \right\|_X ,$$

where $\mathbf{u} = \begin{bmatrix} u & p \end{bmatrix}^T$. Notice that we can rewrite the quantity inside the norm into a matrix-vector product $F\alpha$ where $F = \{\tilde{\mathbf{u}}_{j+1} - \mathbf{u}_j\}_{j=k-m}^{k}$ and $\alpha = \{\alpha_j\}_{j=k-m}^{m}$. We may rewrite this system to an $(m+1) \times m$ linear system to include the $\sum_{j=k-m}^{k} \alpha_{j=k-m}^{k} = 1$ constraint. Let $\{F_i\}_{j=k-m}^{k}$ be the columns of F, the new system is

$$\min \left\| \begin{bmatrix} F_{k-m} & \dots & F_{k-1} & F_k \\ 1 & \dots & 1 & 1 \end{bmatrix} \begin{bmatrix} \alpha_{k-m}^{k+1} \\ \vdots \\ \alpha_{k-1}^{k+1} \\ \alpha_k^{k+1} \end{bmatrix} - \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \right\|_X .$$

However, this is an overdetermined system and most likely will not precisely give $\sum_{j=k-m}^{k} \alpha_j^{k+1} = 1$. So we will rewrite the problem,

$$\begin{aligned}
\min_{\sum_{j=k-m}^{k} \alpha_j^{k+1}=1} ||F\alpha||_X^2 &= \min_{\sum_{j=k-m}^{k} \alpha_j^{k+1}=1} \left|\left|F_{k-m}\alpha_{k-m}^{k+1} + \cdots + F_{k-1}\alpha_{k-1}^{k+1} + F_k\alpha_k^{k+1}\right|\right|_X^2 \\
&= \min_{\alpha} \left|\left|F_{k-m}\alpha_{k-m}^{k+1} + \cdots + F_{k-1}\alpha_{k-1}^{k+1} + F_k\left(1 - \sum_{j=k-m}^{k-1} \alpha_j^{k+1}\right)\right|\right|_X^2 \\
&= \min_{\alpha} \left|\left|(F_{k-m} - F_k)\alpha_{k-m}^{k+1} + \cdots + (F_{k-1} - F_k)\alpha_{k-1}^{k+1} + F_k\right|\right|_X^2 \\
&= \min_{\hat{\alpha}} \left|\left|\hat{F}\hat{\alpha} + F_k\right|\right|_X^2,
\end{aligned}$$

where $\{F_j - F_k\}_{j=k-m}^{k-1}$ are the columns of $\hat{F}$ and $\left\{\alpha_j^{k+1}\right\}_{j=k-m}^{k-1}$ are the entries of $\hat{\alpha}$. This can be seen as a standard least squares problem with the X norm. We solve this in deal.II using the normal equations, i.e.,

$$\hat{\alpha} = -(\hat{F}^T M\hat{F})^{-1}\hat{F}^T MF_k, \tag{6.3}$$

where M is the matrix given from the norm X. M is the stiffness matrix in H_0^1, the mass matrix in L^2, and the identity matrix in ℓ^2. After we solve for $\hat{\alpha}$, we need only solve for α_k^{k+1}, which is a trivial computation. Analysis in [70] uses the H_0^1 norm, but it is unclear if this is needed instead of the less expensive ℓ^2 norm.

The computation of (6.3) is the most costly part of the AA implementation because of the building of $\hat{F}^T M\hat{F}$. As k increases, $\hat{F}$ increases width, which makes building $\hat{F}^T M\hat{F}$ more costly. $\hat{F}^T M\hat{F}$ is a small matrix of size $k \times k$ where k will likely not realistically exceed 20. We note that $\left\{\hat{F}^T M\hat{F}\right\}_{i,j=1}^{k} = (\hat{F}_j^T M)\hat{F}_i$, which is a matrix-vector product and a dot product for each entry. $\hat{F}^T MF_k$ is two matrix-vector products, and since $\hat{F}^T M\hat{F}$ is small, we use an invert function. In total, we perform k matrix-vector products and k^2 dot products. We must also store $2k$ vectors for the $\hat{F}^T M$ computation. In the case where we use the ℓ^2 norm, we skip the $\hat{F}^T M$ calculation and perform a dot product $\hat{F}^T\hat{F}$, thus avoiding k matrix-vector product for the entire $\hat{\alpha}$ calculation and we do not need to store any additional vectors.

6.1.3 Numerical Results

In the previous section, we discussed the AA implementation in deal.II, and so the next step is to integrate it into one of its tutorial programs. Step-57 is a standard benchmark problem where we solve the steady NSE with adaptive refinement. The focus of the tutorial is show the rate of convergence of Newton iterations with line search and the effect of changing values of Re has on it. Here, we substitute the Newton iteration with line search [40] with a Picard iteration with Anderson acceleration.

Note that (6.3) uses the matrix associated with the norm of the operator. In fact, [70] contains analysis using the standard H_0^1 norm for the NSE with Picard iteration. To calculate $\hat{\alpha}$ using the ℓ^2 norm is the fastest, however that is not the norm which the literature has shown in its analysis.

We provide tables which include the number of iterations until convergence with a tolerance of 10^{-13} along with the time in seconds for the entire computation of the program. We also include a table that includes time spent on the AA phase. We ran using $Re = 1, 10, 100, 1000, 2500$ and $m = 0, 1, 2, 10$. These tables will be grouped by which norm was used to perform the minimization step, so we will have a table for each of the H_0^1 norm, L^2 norm, and ℓ^2 norm. This was run on a 64×64 mesh with 37507 DoFs and Taylor-Hood (Q_2, Q_1) elements. GMRES was used to solve the linear system with a block preconditioner using Schur complement. Grad-div stabilization was used with $\gamma = 1$.

Iterations	$m = 0$	$m = 1$	$m = 2$	$m = 10$
$Re = 1$	4	4	4	4
$Re = 10$	6	6	6	7
$Re = 100$	14	14	13	16
$Re = 1000$	37	32	29	22
$Re = 2500$	51	43	39	28

Time	$m = 0$	$m = 1$	$m = 2$	$m = 10$
$Re = 1$	4.8	4.8	4.8	4.8
$Re = 10$	5.4	5.4	5.4	6.2
$Re = 100$	9.7	9.8	9.1	11.0
$Re = 1000$	23.4	20.4	18.7	14.8
$Re = 2500$	32.1	27.4	25.3	19.0

Table 6.1: Iteration count (left) and computation time in seconds (right) with the H_0^1 norm applied to the AA minimization step where $m = 0$ is Picard.

We notice the AA being particularly helpful for $Re \geq 1000$, where setting $m = 10$ lets it converge in 15-20 iterations less than normal Picard without AA ($m = 0$). Something to note is that when $m = 10$ with $Re = 100$, we get the longest convergence at 16 iterations, where $m = 0, 1, 2$ are 14 iterations or less. It seems that applying low values for m help more with lower values for Re, and higher m is helpful for higher Re.

The same could be said about the computation time. For $Re \leq 100$, we have that $m = 2$ give the best results for both time (although the difference is not too momentous). Applying $m = 10$ has the highest computation time for $Re = 100$, which is because it takes more iterations. Then with $Re = 2500$, we see that $m = 10$ is completed in 19 seconds, where Picard takes 32.1 seconds.

Iterations	$m = 0$	$m = 1$	$m = 2$	$m = 10$
$Re = 1$	4	4	4	4
$Re = 10$	6	6	6	7
$Re = 100$	14	14	13	16
$Re = 1000$	37	30	27	22
$Re = 2500$	51	44	37	27

Time	$m = 0$	$m = 1$	$m = 2$	$m = 10$
$Re = 1$	4.9	4.9	4.7	4.7
$Re = 10$	5.4	5.4	5.4	6.2
$Re = 100$	9.7	9.7	9.1	11.1
$Re = 1000$	23.2	19.3	17.5	14.7
$Re = 2500$	31.8	27.8	23.8	18.0

Table 6.2: Iteration count (left) and computation time in seconds (right) with the L^2 norm applied to the AA minimization step

There is a substantial difference in iteration count for $Re \geq 1000$. In particular, using AA with $m = 1, 2$ for $Re = 1000$ converges in two fewer iterations than using the H_0^1 norm. We also have fewer iterations for $m = 2, 10$ for $Re = 2500$, however iteration count for $m = 1$ has increased from 43 with the H_0^1 norm to 44 with the L^2 norm.

We see similar computation times with L^2 than H_0^1. Similar comments can be made here, where $m = 2$ gave the better computation time for $Re \leq 100$ and $m = 10$ gave the best results for $Re \geq 1000$.

Iterations	$m = 0$	$m = 1$	$m = 2$	$m = 10$
$Re = 1$	4	4	4	4
$Re = 10$	6	6	6	7
$Re = 100$	14	14	13	16
$Re = 1000$	37	30	27	21
$Re = 2500$	51	44	37	27

Time	$m = 0$	$m = 1$	$m = 2$	$m = 10$
$Re = 1$	4.7	4.9	4.7	4.7
$Re = 10$	5.3	5.3	5.3	6.0
$Re = 100$	9.6	9.6	8.9	10.8
$Re = 1000$	23.1	19.1	17.7	14.1
$Re = 2500$	32.4	27.9	23.8	18.0

Table 6.3: Iteration count (left) and computation time in seconds (right) with the ℓ^2 norm applied to the AA minimization step.

We see minor improvement with the ℓ^2 norm. The iteration count is nearly identical to the L^2 norm save for one experiment. We expect to have a better computation time with the ℓ^2 norm because $M = I$. We only see an improvement of one experiment where $Re = 1000$ with $m = 10$ converges in 21 iterations rather than 22 iterations with the L^2 norm.

Similar comments can be made how the computation time is affected for $Re \geq 1000$. We see that ℓ^2 with $Re = 1000$ we have an improvement of computation time for all but when $m = 2$ ($m = 0$ can be ignored because it does not include the AA step). To understand the minor improvement in computation time in Figure 6.3, we will include one more table which includes the computation time in seconds of the AA step.

	H_0^1 norm			L^2 norm			ℓ^2 norm		
AA times	$m=1$	$m=2$	$m=10$	$m=1$	$m=2$	$m=10$	$m=1$	$m=2$	$m=10$
$Re=1$	0.01	0.01	0.02	0.01	0.01	0.02	0.00	0.00	0.01
$Re=10$	0.01	0.02	0.05	0.01	0.02	0.05	0.00	0.01	0.02
$Re=100$	0.04	0.05	0.22	0.04	0.05	0.23	0.02	0.01	0.06
$Re=1000$	0.08	0.11	0.35	0.08	0.10	0.35	0.02	0.03	0.09
$Re=2500$	0.11	0.15	0.48	0.11	0.14	0.44	0.03	0.04	0.03

Table 6.4: Time in seconds spent on AA.

The H_0^1 norm and the L^2 norm provide similar results. This is expected because the k matrix-vector products using the mass and stiffness matrix will take a very similar amount of time to compute. Hence we do not need to comment on the differences between these two norms.

As we increase Re, we see an expected increase in the time spent in AA among each experiment because the iteration count is increasing. However for H_0^1 and L^2, we notice a larger increase in time from $m = 1$ to $m = 10$. This is expected because of the $\hat{F}^T M \hat{F}$ calculation. In other words this is the result from doing one matrix-vector product and one dot product ($k = 1$) to 10 matrix-vector products and 100 dot products ($k = 10$).

Here we see the strength of the ℓ^2 norm over the other two. In the most extreme case, $Re = 2500$ and $m = 10$, we see the time spent in AA is 0.03 seconds, which is a major improvement over the 0.48 we see with H_0^1. Based on this, it is clear that the ℓ^2 is the most advantageous matrix norm to compute with.

6.2 Newton comparison and Higher Re experiments

Our motivation for implementing AA in deal.II is to have a faster nonlinear solver than what is available. Here we will compare results received from step-57 and compare it to what we get with our AA implementation. For this first experiment, we will be comparing our results in the previous section with the ℓ^2 norm with the built-in Newton iterations in step-57. The setup for the problem is the same as the previous experiment, i.e., 64 × 64 mesh with 37507 DoFs. We also modify the continuation method for Newton to converge with a finer mesh. We do one continuation step for every increase by 500 in Re, so with $Re = 2500$, we do 5 continuation steps.

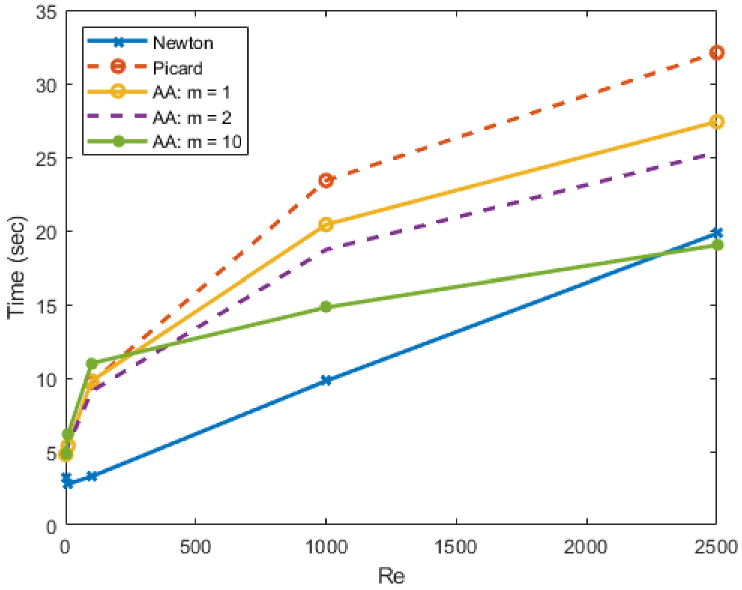

Figure 6.1: Computation time (seconds) comparison between AA implementation and Newton with line search on a 64 × 64 mesh.

In Figure 6.1, Newton has a steady increase in computation time as Re increases. A Newton solve is performed every iteration in the continuation method, which takes a consistent amount of time, hence the somewhat linear behavior of Newton. Newton outperforms both Picard and AA consistently until $Re = 2500$. It is here where AA with $m = 10$ is the only method that is an improvement over Newton.

Our next experiment is solving with much higher Re on a 128 × 128 mesh, which is required with the higher Re. Below is a table containing the computation times for our algorithm and the algorithm given by step-57. It should be noted that running this simulation with $Re > 8000$ is less physically meaningful because this solution is no longer stationary [7]. We feel that this does not take away from our goal of showing how AA is an improvement over Newton with high Re.

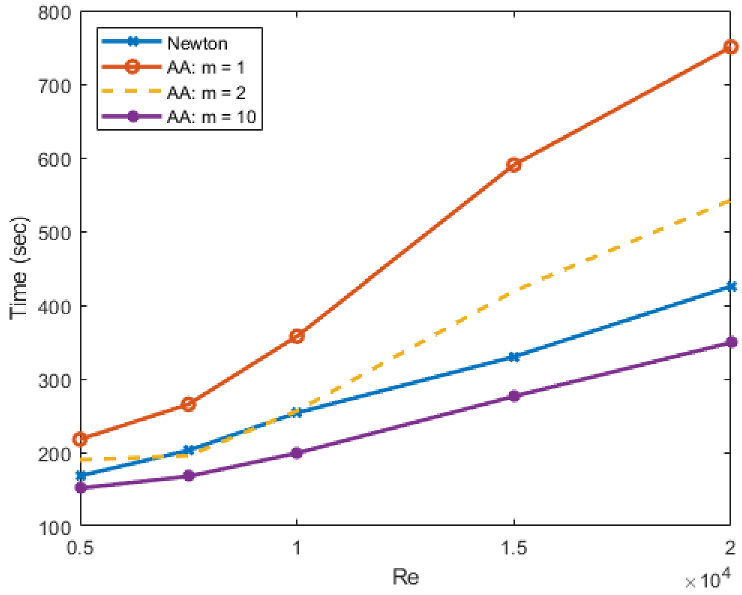

Figure 6.2: Computation time (seconds) comparison between AA implementation and Newton with line search with continuation step of 500 for Re on a 128 × 128 mesh.

According to Figure 6.2, AA with $m = 1$ was consistently the worst option for this experiment. In fact as Re increases, the gap between computation time of other methods and AA with $m = 1$ increases as well. Continuing from Figure 6.1, AA with $m = 10$ has the fastest computation time for all values of Re. We also include a table of iteration counts for this experiment (excluding Newton).

Iterations	$m = 1$	$m = 2$	$m = 10$
$Re = 5000$	57	49	38
$Re = 7500$	70	51	43
$Re = 10000$	94	66	50
$Re = 15000$	153	110	72
$Re = 20000$	196	141	89

Table 6.5: Iteration count for our AA algorithm on a 128 × 128 mesh.

The results shown in Table 6.5 are consistent with what we have seen so far with tables 6.1-6.3. With $Re \geq 10000$, increasing m from 1 to 10 has cut the iteration count in half.

6.3 KINSOL implementation to step-57

KINSOL is a solver for nonlinear algebraic systems developed by Lawrence Livermore National Laboratory which includes Newton, Picard, and fixed point solvers with AA options. KINSOL is a subcode of SUNDIALS [47], which is as its namesake defines a "SUite of Nonlinear and DIfferential/ALgebraic equation Solvers". Other codes under the umbrella of SUNDIALS include: CVODE, which solves initial value problems for ODE systems, ARKODE, which solves initial value ODE problems with Runge-Kutta methods, IDA, which solves initial value problems for differential-algebraic systems, and more.

The integration of SUNDIALS is a more recent addition in deal.II. The tutorial program step-77 was written by Wolfgang Bangerth within the last couple years and it has yet to see thorough experimentation in deal.II. In this section we will integrate the KINSOL nonlinear solver into step-57 and compare results to Sections 6.1 and 6.2.

The implementation for KINSOL is straightforward if one is familiar with how deal.II handles linear system assembly. Take this code block from the `assembly()` function from step-57:

```
for (unsigned int i = 0; i < dofs_per_cell; ++i)
{
  if (assemble_matrix)
  {
    for (unsigned int j = 0; j < dofs_per_cell; ++j)
    {
      local_matrix(i, j) +=
        (viscosity *
           scalar_product(grad_phi_u[j], grad_phi_u[i]) +
         present_velocity_gradients[q] * phi_u[j] * phi_u[i] +
         grad_phi_u[j] * present_velocity_values[q] *
           phi_u[i] -
         div_phi_u[i] * phi_p[j] - phi_p[i] * div_phi_u[j] +
         gamma * div_phi_u[j] * div_phi_u[i] +
         phi_p[i] * phi_p[j]) *
        fe_values.JxW(q);
     }
  }

  double present_velocity_divergence =
    trace(present_velocity_gradients[q]);
  local_rhs(i) +=
    (-viscosity * scalar_product(present_velocity_gradients[q],
                                 grad_phi_u[i]) -
     present_velocity_gradients[q] * present_velocity_values[q] *
       phi_u[i] +
     present_pressure_values[q] * div_phi_u[i] +
     present_velocity_divergence * phi_p[i] -
     gamma * present_velocity_divergence * div_phi_u[i]) *
    fe_values.JxW(q);
}
```

Here we calculate the system matrix and the system right hand side vector in the same loop. KINSOL uses lambda functions in C++ to define specific functions. Our problem that we are solving requires four functions:

- `reinit_vector()`, vector re-initialization.
- `residual()`, calculating the right hand side vector, $f - G(u_k)u_k$ in our case.
- `setup_jacobian()`, compute the system matrix.
- `solve_with_jacobian()`, main linear solver function.

KINSOL requires we compute the system matrix and the residual in separate functions, so we simply need to separate the `local_matrix(i,j)` and `local_rhs(i)` calculations to different functions. Aside from small differences like this, the program is very similar and follows the same logic.

We run a similar experiment to that of Sections 6.1-6.2. We are computing on a 64 × 64 grid with Taylor-Hood (Q_2, Q_1) elements. We solve the linear system using GMRES with a block preconditioner using Schur complement. We use an initial guess of the zero vector, similar to the algorithm we implemented in section 6.1. Below is a table of computation time.

Computation time	Line Search	Picard	AA: $m = 1$	AA: $m = 2$	AA: $m = 10$
$Re = 1$	3.9	3.6	3.6	3.6	3.6
$Re = 10$	5.0	4.5	4.5	4.5	4.5
$Re = 100$	7.5	8.4	8.4	8.4	7.9
$Re = 1000$	35.2	20.8	20.2	17.0	14.1
$Re = 2500$	-	32.1	28.8	24.7	19.1

Table 6.6: Computation time using KINSOL.

Including every experiment we have done to this point, KINSOL gives the best results for $Re \leq 100$. In Figure 6.3, we have computation times of our algorithm using the ℓ^2 norm. For $Re = 100$, we have our fastest convergence of AA with $m = 2$ at 8.9 seconds. KINSOL gives us our slowest convergence of 8.4 seconds (Picard, AA with $m = 1, 2$) and applying AA with $m = 10$ gives our fastest result at 7.9 seconds, nearly 3 seconds faster than the same experiment with our algorithm.

For $Re \geq 1000$, our algorithm gives us favorable results for AA. For $Re = 1000$ and $m = 1$, our algorithm gives a computation time of 19.1 seconds where KINSOL gives 20.2 seconds. With $m = 2$, KINSOL is faster with 17.0 seconds, then with $m = 10$ the computation times are the same. $Re = 2500$ is where our algorithm performed the best where AA was faster in each experiment. For $m = 1, 2$, our algorithm was 0.9 seconds faster and for $m - 10$ our algorithm was 1.1 seconds faster.

Among all nonlinear solvers studied in this chapter, KINSOL has given us the fastest convergence for $Re \leq 100$ including the line search algorithm in step-57. If one were to solve the nonlinear problem for the SNSE (and the NSE by extension), it would be optimal to use KINSOL for flows with low Re. When Re increases, KINSOL and our implementation have more similar convergence times. The deciding factor for using another package or using our algorithm seems to come down to how the code is written or whether one wants to download the SUNDIALS package. Using KINSOL requires one to rewrite the structure of standard deal.II code designed from the example programs. With our algorithm, we write two extra functions and change a couple lines in the assembly. Based on these reasons and our results, the optimal choice would come down to user preference.

Chapter 7

Conclusion and future projects

In this dissertation, we introduced a new energy, momentum, and angular momentum conserving scheme with a spatial filter, EMAC-Reg. We have proven that it is a stable, well-posed scheme and provided an error analysis. To show EMAC-Reg's strengths, we compared it to other numerical schemes on coarser meshes. In every case, we have that EMAC-Reg is a stronger model with coarser meshes. This is advantageous for large scale computing and 3D domains, where the linear systems become colossal.

We have also introduced the idea of FOM-ROM consistency between models. We provided an error analysis under the assumption that a FOM-ROM pair is inconsistent. The preeminent result is the existence of an extra error term, which would not be found in a standard FOM-ROM consistent system. It is for this reason that we conclude that having FOM-ROM inconsistency is an overall destructive quality. We have provided a numerical experiment that reinforce this claim.

In the following chapter, we provided fully discrete error analyses for SKEW and EMAC. In particular, we highlighted the fact that under certain conditions for EMAC, the Gronwall constant does not depend on *Re*. This is helpful because the error bound will grow exponentially for larger values of *Re*. We provided several numerical experiments which include projection methods for comparison and found that EMAC consistently outperformed its competitors.

Finally, we have included an implementation of Anderson acceleration into the software library deal.II, which was based on the example program step-57. We provided specific details on how the optimization step is performed and how we integrated different matrix norms into the computation. We provided several numerical results with details on iteration count and computation time. We compared our implementation to the base program step-57, which uses Newton iteration with linesearch. Finally, we compared our results to the nonlinear solver KINSOL. Here, we discovered KINSOL is best applied for problems with lower *Re*, but the results are comparable to our algorithm for higher *Re*. The optimal method of solving the nonlinear problem comes down to user preference on code design.

7.1 Future work

A future direction to take EMAC-Reg is large scale computing and 3D FEM implementation. The motivation for using EMAC-Reg is that 3D computing can require unrealistically (with modern computers) fine meshes, which is where EMAC-Reg can be very helpful. There has been extensive study on the behavior of CONV and SKEW (and EMAC to an extent), and to show the utility of EMAC-Reg on a coarser 3D mesh would yield useful results.

We need not only limit future direction to EMAC-Reg, any combination of topics on this dissertation can be applied to different multiphysics problems, e.g., Boussinesq approximation. It is worth studying the application of ROM models and EMAC-Reg in other settings to understand and compare how these models compare to already known and established methods.

Expanding the capabilities of nonlinear solvers in deal.II is also an interesting direction. We have shown the strengths of applying Anderson acceleration to problems over other methods in Chapter 6. We discussed the strengths of some nonlinear solve methods over others, e.g., KINSOL is stronger with low *Re* and the algorithm we developed has better convergence for large *Re*. It would be a nice addition to deal.II to implement a nonlinear solver class that uses different solvers and methods depending on the problem parameters.

Appendices

Appendix A Momentum/angular momentum conservation of NS-α and Leray-α formulations

Here we show that the NS-α and Leray-α formulations do not conserve momentum or angular momentum if $\nabla \cdot u \neq 0$ and $\nabla \cdot w \neq 0$ where w represents the filtered velocity $\bar{u}$. Further note $u = Fw$ where $F = -\alpha^2 \Delta I + I$.

A.1 NS-α

Recall the nonlinear term of the NS-α formulation is

$$u_t + (\nabla \times u) \times w + \nabla p - \nu \Delta u = f. \tag{1}$$

Test (1) with e_i for $i = 1, 2, 3$. After applying the space-time divergence theorem and rearranging some we get

$$(u_t, e_i) + ((\nabla \times u) \times w, e_i) + \nu (\nabla u, \nabla e_i) = (f, e_i). \tag{2}$$

Assuming $\nu = f = 0$, (2) simplifies into

$$\frac{d}{dt}(u, e_i) + ((\nabla \times u) \times w, e_i) = 0.$$

If the nonlinear term is equal to zero, then we will have momentum conservation. We now check this:

$$\begin{aligned} ((\nabla \times u) \times w, e_i) &= (\nabla \times (w \times e_i), u) \\ &= ((\nabla \cdot e_i) w, u) - ((\nabla \cdot w) e_i, u) + (e_i \cdot \nabla w, u) - (w \cdot \nabla e_i, u), \end{aligned}$$

where the above two equalities come from vector identities. Also note that because e_i is a vector of scalars, $((\nabla \cdot e_i) w, u) = (w \cdot \nabla e_i, u) = 0$. This leaves us with

$$((\nabla \times u) \times w, e_i) = -((\nabla \cdot w) e_i, u) + (e_i \cdot \nabla w, u),$$

which we cannot conclude is zero, hence we cannot say that the NS-α formulation preserves momentum.

For angular momentum, we test (1) with ϕ_i and the algebra works out similar to momentum,

$$((\nabla \times u) \times w, \phi_i) = ((\nabla \cdot \phi_i)w, u) - ((\nabla \cdot w)\phi_i, u) + (\phi_i \cdot \nabla w, u) - (w \cdot \nabla \phi_i, u).$$

Since $\nabla \cdot \phi_i = 0$ for $i = 1, 2, 3$, we have

$$((\nabla \cdot \phi_i)w, u) = 0.$$

Also recall using (4.15) in Theorem 11, we have

$$(w \cdot \nabla \phi_i, u) = 0.$$

This gives us

$$((\nabla \times u) \times w, \phi_i) = -((\nabla \cdot w)\phi_i, u) + (\phi_i \cdot \nabla w, u).$$

Much like with momentum, we cannot conclude that this quantity is zero, and we expect it is not zero.

A.2 Leray-α

Recall the nonlinear term of the Leray-α formulation is

$$u_t + w \cdot \nabla u + \nabla p - \nu \Delta u = f. \tag{3}$$

We test (3) with e_i for $i = 1, 2, 3$ and integrate. Similar to (2)

$$(u_t, e_i) + (w \cdot \nabla u, e_i) + \nu (\nabla u, \nabla e_i) = (f, e_i). \tag{4}$$

Assuming $\nu = f = 0$, (4) simplifies to

$$\frac{d}{dt}(u, e_i) + (w \cdot \nabla u, e_i) = 0.$$

If the nonlinear term is equal to zero, then we will have momentum conservation. Using (2.7) on the nonlinear term we get

$$\begin{aligned}(w \cdot \nabla u, e_i) &= -(w \cdot \nabla e_i, u) - ((\nabla \cdot w)u, e_i) \\ &= -((\nabla \cdot w)u, e_i),\end{aligned}$$

which is not zero when $\nabla \cdot w \neq 0$. Hence momentum is not necessarily conserved.

For angular momentum we test (3) with ϕ_i for $i = 1, 2, 3$ and it simplifies to

$$\frac{d}{dt}(u, \phi_i) + (w \cdot \nabla u, \phi_i) = 0.$$

Now similarly to the momentum proof, we have for the nonlinear term

$$\begin{aligned}(w \cdot \nabla u, \phi_i) &= -(w \cdot \nabla \phi_i, u) - ((\nabla \cdot w)u, \phi_i) \\ &= -((\nabla \cdot w)u, \phi_i),\end{aligned}$$

where the first term disappears by applying (4.15) similarly to Theorem 11 (and the angular momentum proof for NS-α in Appendix A.1). Thus the nonlinear term does not vanish, so angular momentum is not conserved.

Appendix B Link to code

Our implementation of Anderson acceleration and our implementation of KINSOL to step-57 of deal.II can be found with the following link

`https://github.com/singima/Grad_Research/tree/main/Dissertation_Programs`

Bibliography

[1] S. A. Huyer A. M. Dropkin and C. Henoch. Combined Experimental/Numerical Development of Propulsor Evaluation Capability. *Journal of Fluids Engineering*, 133(8), 09 2011. 081105.

[2] S. E. Ahmed, S. Pawar, O. San, A. Rasheed, T. Iliescu, and B. R. Noack. On closures for reduced order models—a spectrum of first-principle to machine-learned avenues. *Physics of Fluids*, 33(9):091301, 2021.

[3] H. An, X. Jia, and H. F. Walker. Anderson acceleration and application to the three-temperature energy equations. *Journal of Computational Physics*, 347:1–19, 2017.

[4] D. G. Anderson. Iterative procedures for nonlinear integral equations. *Journal of the ACM (JACM)*, 12(4):547–560, 1965.

[5] D. Arndt, W. Bangerth, M. Feder, M. Fehling, R. Gassmöller, T. Heister, L. Heltai, M. Kronbichler, M. Maier, P. Munch, et al. The deal.II library, version 9.4. *Journal of Numerical Mathematics*, 30(3):231–246, 2022.

[6] D. Arnold and J. Qin. Quadratic velocity/linear pressure Stokes elements. In R. Vichnevetsky, D. Knight, and G. Richter, editors, *Advances in Computer Methods for Partial Differential Equations VII*, pages 28–34. IMACS, 1992.

[7] W. Bangerth and R. Rannacher. Adaptive finite element techniques for the acoustic wave equation. *Journal of Computational Acoustics*, 9(02):575–591, 2001.

[8] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.

[9] L.C. Berselli, T. Iliescu, and W.J. Layton. *Mathematics of large eddy simulation of turbulent flows*. Scientific Computation. Springer-Verlag, Berlin, 2006.

[10] A. Both, O. Lehmkuhl, D. Mira, and M. Ortega. Low-dissipation finite element strategy for low mach number reacting flows. *Computers & Fluids*, 200:104436, 2020.

[11] A. Bowers, T.-Y. Kim, M. Neda, L. Rebholz, and E. Fried. The Leray-$\alpha\beta$-deconvolution model: energy analysis and numerical algorithms. *Applied Mathematical Modelling*, 37(3):1225–1241, 2013.

[12] J. Bramble, J. Pasciak, and O. Steinbach. On the stability of the L^2 projection in $H^1(\Omega)$. *Math. Comp.*, 71:147–156, 2002.

[13] S. Breckling, M. Neda, and F. Pahlevani. A sensitivity study of the Navier–Stokes-α model. *Computers & Mathematics with Applications*, 75(2):666–689, 2018.

[14] S. Brenner and L. R. Scott. *The Mathematical Theory of Finite Element Methods*. Springer-Verlag, 1994.

[15] S. L. Brunton, J. L. Proctor, and J. N. Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, 2016.

[16] A. Caiazzo, T. Iliescu, V. John, and S. Schyschlowa. A numerical investigation of velocity–pressure reduced order models for incompressible flows. *Journal of Computational Physics*, 259:598–616, 2014.

[17] K. Carlberg, C. Bou-Mosleh, and C. Farhat. Efficient non-linear model reduction via a least-squares Petrov–Galerkin projection and compressive tensor approximations. *International Journal for numerical methods in engineering*, 86(2):155–181, 2011.

[18] M. Case, V. Ervin, A. Linke, and L. Rebholz. A connection between Scott-Vogelius elements and grad-div stabilization. *SIAM Journal on Numerical Analysis*, 49(4):1461–1481, 2011.

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

[20] S. Charnyi, T. Heister, M. Olshanskii, and L. Rebholz. Efficient discretizations for the EMAC formulation of the incompressible Navier-Stokes equations. *Applied Numerical Mathematics*, 141:220–233, 2019.

[21] S. Chen, C. Foias, D.D. Holm, E. Olson, E.S. Titi, and S. Wynne. The Camassa–Holm equations as a closure model for turbulent channel and pipe flow. *Physical Review Letters*, 81:5338–5341, 1998.

[22] S. Chen, C. Foias, E. Olson, E.S. Titi, and W. Wynne. The Camassa–Holm equations and turbulence. *Physica D*, 133:49–65, 1999.

[23] A. Cheskidov, D. Holm, E. Olson, and E.S. Titi. On a Leray-α model of turbulence. *Proceedings of The Royal Society A*, 461:629–649, 2005.

[24] A. J. Chorin. Numerical solution for the Navier-Stokes equations. *Math. Comp.*, 22:745–762, 1968.

[25] J. Connors. Convergence analysis and computational testing of the finite element discretization of the Navier–Stokes α model. *Numerical Methods for Partial Differential Equations*, 26(6):1328–1350, 2010.

[26] V. Cuff, A. Dunca, C. Manica, and L. Rebholz. The reduced order NS-α model for incompressible flow: theory, numerical analysis and benchmark testing. *ESAIM: Mathematical Modelling and Numerical Analysis*, 49(3):641–662, 2015.

[27] A. Dunca and Y. Epshteyn. On the Stolz-Adams deconvolution model for the large-eddy simulation of turbulent flows. *SIAM J. Math. Anal.*, 37(6):1890–1902, 2005.

[28] C. Evans, S. Pollock, L. G. Rebholz, and M. Xiao. A proof that Anderson acceleration improves the convergence rate in linearly converging fixed-point methods (but not in those converging quadratically). *SIAM Journal on Numerical Analysis*, 58(1):788–810, 2020.

[29] J. A. Evans and T. J. R. Hughes. Isogeometric divergence-conforming B-splines for the steady Navier–Stokes equations. *Mathematical Models and Methods in Applied Sciences*, 23(08):1421–1478, 2013.

[30] C. L. Fefferman. Existence and smoothness of the Navier-Stokes equation. *The millennium prize problems*, 57(67):22, 2006.

[31] C. Foias, D.D. Holm, and E.S. Titi. The Navier–Stokes-α model of fluid turbulence. *Physica D*, 152:505–519, 2001.

[32] C. Foias, D.D. Holm, and E.S. Titi. The three dimensional viscous Camassa–Holm equations, and their relation to the Navier–Stokes equations and turbulence theory. *Journal of Dynamics and Differential Equations*, 14:1–35, 2002.

[33] U. Frisch. *Turbulence*. Cambridge University Press, 1995.

[34] M. Geist and B. Scherrer. Anderson acceleration for reinforcement learning. *arXiv preprint arXiv:1809.09501*, 2018.

[35] P. G. Geredeli, L. G. Rebholz, D. Vargun, and A. Zytoon. Improved convergence of the Arrow–Hurwicz iteration for the Navier–Stokes equation via grad–div stabilization and Anderson acceleration. *Journal of Computational and Applied Mathematics*, 422:114920, 2023.

[36] B.J. Geurts and D. Holm. Regularization modeling for large eddy simulation. *Physics of Fluids*, 15:L13, 2003.

[37] B.J. Geurts and D. Holm. Leray and LANS-α modeling of turbulent mixing. *Journal of Turbulence*, 7:1–33, 2006.

[38] V. Girault and P.-A. Raviart. *Finite element methods for Navier-Stokes equations : theory and algorithms*. Springer-Verlag, 1986.

[39] P. Gresho. On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via finite-element method that also introduces a nearly consistent mass matrix: Part 2: Applications. *Int. J. Numer. Methods Fluids*, 11:621?659, 1990.

[40] L. Grippo, F. Lampariello, and S. Lucidi. A nonmonotone line search technique for newton's method. *SIAM journal on Numerical Analysis*, 23(4):707–716, 1986.

[41] J. Guermond, P. Minev, and J. Shen. An overview of projection methods for incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 195:6011–6045, 2006.

[42] J. Guzman and M. Neilan. Conforming and divergence-free Stokes elements in three dimensions. *IMA Journal of Numerical Analysis*, 34(4):1489–1508, 2014.

[43] J. Guzman and M. Neilan. Conforming and divergence-free Stokes elements on general triangular meshes. *Mathematics of Computation*, 83:15–36, 2014.

[44] F. Hecht. New development in FreeFem++. *Journal of Numerical Mathematics*, 20(3-4):251–266, 2012.

[45] J. S. Hesthaven, G. Rozza, B. Stamm, et al. *Certified reduced basis methods for parametrized partial differential equations*, volume 590. Springer, 2016.

[46] J. Heywood and R. Rannacher. Finite element approximation of the nonstationary Navier-Stokes problem. Part IV: Error analysis for the second order time discretization. *SIAM J. Numer. Anal.*, 2:353–384, 1990.

[47] A. Hindmarsh, P. Brown, K. Grant, S. Lee, R. Serban, D. Shumaker, and C. Woodward. Sundials: Suite of nonlinear and differential/algebraic equation solvers. *ACM Transactions on Mathematical Software (TOMS)*, 31(3):363–396, 2005.

[48] D. Holm and B.T. Nadiga. Modeling mesoscale turbulence in the barotropic double-gyre circulation. *Journal of Physical Oceanography*, 33:2355–2365, 2003.

[49] P. Holmes, J. L. Lumley, G. Berkooz, and C. W. Rowley. *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge university press, 2012.

[50] S. A. Huyer. Application of a maneuvering propulsor technology to undersea vehicles. *Journal of Fluids Engineering*, 136(1), 10 2013. 011103.

[51] S. Ingimarson. An energy, momentum, and angular momentum conserving scheme for a regularization model of incompressible flow. *Journal of Numerical Mathematics*, 30(1):1–22, 2022.

[52] S. Ingimarson, M. Neda, L. G. Rebholz, J. Reyes, and A. Vu. Improved long time accuracy for projection methods for Navier-Stokes equations using EMAC formulation. *International Journal of Numerical Analysis and Modeling*, 20(2):176–198, 2023.

[53] S. Ingimarson, L. Rebholz, and T. Iliescu. Full and reduced order model consistency of the nonlinearity discretization in incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 401:115620, 2022.

[54] 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*, 44:777–788, 2004.

[55] V. John, A. Linke, C. Merdon, M. Neilan, and L. G. Rebholz. On the divergence constraint in mixed finite element methods for incompressible flows. *SIAM Rev.*, 59(3):492–544, 2017.

[56] B. Koc, S. Rubino, M. Schneier, J. Singler, and T. Iliescu. On optimal pointwise in time error bounds and difference quotients for the proper orthogonal decomposition. *SIAM Journal on Numerical Analysis*, 59(4):2163–2196, 2021.

[57] A.N. Kolmogorov. Dissipation of energy in isotropic turbulence. *Dokl. Akad. Nauk. SSSR*, 32:19–21, 1941.

[58] A.N. Kolmogorov. The local structure of turbulence in incompressible viscous fluids at very large Reynolds numbers. *Dokl. Akad. Nauk. SSSR*, 30:299–303, 1941.

[59] A.N. Kolmogorov. On the degeneration of isotropic turbulence in an incompressible viscous fluids. *Dokl. Akad. Nauk. SSSR*, 31:538–541, 1941.

[60] K. Kunisch and S. Volkwein. Galerkin proper orthogonal decomposition methods for parabolic problems. *Numerische mathematik*, 90(1):117–148, 2001.

[61] W. Layton. *Introduction to the Numerical Analysis of Incompressible Viscous Flows*. SIAM, 2008.

[62] W. Layton, C. Manica, M. Neda, and L. Rebholz. Numerical analysis and computational testing of a high accuracy Leray-deconvolution model of turbulence. *Numerical Methods for Partial Differential Equations*, 24(2):555–582, 2008.

[63] W. Layton and L. Rebholz. *Approximate Deconvolution Models of Turbulence: Analysis, Phenomenology and Numerical Analysis*. Springer-Verlag, 2012.

[64] O. Lehmkuhl, G. Houzeaux, H. Owen, G. Chrysokentis, and I. Rodríguez. A low-dissipation finite element scheme for scale resolving simulations of turbulent flows. *Journal of Computational Physics*, 390:51–65, 2019.

[65] R. Liska and B. Wendroff. Comparison of several difference schemes on 1D and 2D test problems for the Euler equations. *SIAM Journal on Scientific Computing*, 25:995–1017, 2003.

[66] P.A. Lott, H.F. Walker, C.S. Woodward, and U.M. Yang. An accelerated Picard method for nonlinear systems related to variably saturated flow. *Advances in Water Resources*, 38:92–101, 2012.

[67] V. Mai and M. Johansson. Anderson acceleration of proximal gradient methods. In *International Conference on Machine Learning*, pages 6620–6629. PMLR, 2020.

[68] M. Olshanskii and L. Rebholz. Longer time accuracy for incompressible Navier-Stokes simulations with the EMAC formulation. *Computer Methods in Applied Mechanics and Engineering*, 372:113369, 2020.

[69] A. L. Pavlov, G. W. V. Ovchinnikov, D. Derbyshev, D. Tsetserukou, and I. V. Oseledets. AA-ICP: Iterative closest point with Anderson acceleration. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3407–3412. IEEE, 2018.

[70] S. Pollock, L. G. Rebholz, and M. Xiao. Anderson-accelerated convergence of Picard iterations for incompressible navier–stokes equations. *SIAM Journal on Numerical Analysis*, 57(2):615–637, 2019.

[71] S. Pope. *Turbulent Flows*. Cambridge University Press, 2000.

[72] F. A. Potra and H. Engler. A characterization of the behavior of the Anderson acceleration on linear problems. *linear Algebra and its Applications*, 438(3):1002–1011, 2013.

[73] P. P. Pratapa, P. Suryanarayana, and J. E. Pask. Anderson acceleration of the Jacobi iterative method: An efficient alternative to Krylov methods for large, sparse linear systems. *Journal of Computational Physics*, 306:43–54, 2016.

[74] L. Rebholz. Conservation laws of turbulence models. *Journal of Mathematical Analysis and Applications*, 326(1):33–45, 2007.

[75] L. Rebholz, T.-Y. Kim, and Y.-L. Byon. On an accurate α model for coarse mesh turbulent channel flow simulation. *Applied Mathematical Modelling*, 43:139–154, 2017.

[76] T. C. Rebollo and R. Lewandowski. *Mathematical and numerical foundations of turbulence models and applications.* Springer, 2014.

[77] M. Schäfer and S. Turek. The benchmark problem 'flow around a cylinder' flow simulation with high performance computers II. *in E.H. Hirschel (Ed.), Notes on Numerical Fluid Mechanics*, 52, Braunschweig, Vieweg:547–566, 1996.

[78] P. Schroeder, V. John, P. Lederer, C. Lehrenfeld, G. Lube, and J. Schoberl. On reference solutions and the sensitivity of the 2D Kelvin-Helmholtz instability problem. *Computers and Mathematics with Applications*, 77(4):1010–1028, 2019.

[79] P. Schroeder, C. Lehrenfeld, A. Linke, and G. Lube. Towards computable flows and robust estimates for inf-sup stable FEM applied to the time dependent incompressible Navier-Stokes equations. *SeMA*, 75:629–653, 2018.

[80] P. Schroeder and G. Lube. Pressure-robust analysis of divergence-free and conforming FEM for evolutionary incompressible Navier-Stokes flows. *Journal of Numerical Mathematics*, 25(4):249–276, 2017.

[81] P. W. Schroeder and G. Lube. Divergence-free H (div)-fem for time-dependent incompressible flows with applications to high Reynolds number vortex dynamics. *Journal of Scientific Computing*, 75(2):830–858, 2018.

[82] J. R. Singler. New POD error expressions, error bounds, and asymptotic results for reduced order models of parabolic PDEs. *SIAM Journal on Numerical Analysis*, 52(2):852–876, 2014.

[83] J. S. Smagorinsky. General circulation experiments with the primitive equations. *Mon. Weather Rev.*, 91:99–164, 1963.

[84] A. Sommerfeld. *Ein beitrag zur hydrodynamischen erklaerung der turbulenten fluessigkeitsbewegungen.* 1909.

[85] H. Sterck and Y. He. On the asymptotic linear convergence speed of Anderson acceleration, Nesterov acceleration, and nonlinear GMRES. *SIAM Journal on Scientific Computing*, 43(5):S21–S46, 2021.

[86] G. Stokes et al. On the effect of the internal friction of fluids on the motion of pendulums. 1851.

[87] S. Ray T. Tezduyar, S. Mittal and R. Shih. Incompressible flow computations with stabilized bilinear and linear equal order interpolation velocity-pressure elements. *Computer Methods in Applied Mechanics and Engineering*, 95:221–242, 1992.

[88] A. G. Taylor and A. C. Hindmarsh. User documentation for KINSOL, a nonlinear solver for sequential and parallel computers. *Rapport technique UCRL-ID-131185, Lawrence Livermore National Laboratory*, page 6, 1998.

[89] R. Temam. Sur la stabilité et la convergence de la méthode des pas fractionnaires. *Annali di Matematica pura ed applicata*, 79(1):191–379, 1968.

[90] R. Temam. *Navier-Stokes equations.* Elsevier, North-Holland, 1991.

[91] A. Toth and C. T. Kelley. Convergence analysis for Anderson acceleration. *SIAM Journal on Numerical Analysis*, 53(2):805–819, 2015.

[92] A. Toth, C.T. Kelley, S. R. Slattery, S. P. Hamilton, K. T. Clarno, and R.P.P. Pawlowski. Analysis of Anderson acceleration on a simplified neutronics/thermal hydraulics system. Technical report, Oak Ridge National Lab.(ORNL), Oak Ridge, TN (United States). Consortium for . . . , 2015.

[93] H. F. Walker and P. Ni. Anderson acceleration for fixed-point iterations. *SIAM Journal on Numerical Analysis*, 49(4):1715–1735, 2011.

[94] T. Wu. A study of convergence on the newton-homotopy continuation method. *Applied Mathematics and Computation*, 168(2):1169–1174, 2005.

[95] X. Xie, D. Wells, Z. Wang, and T. Iliescu. Numerical analysis of the Leray reduced order model. *Journal of Computational and Applied Mathematics*, 328:12–29, 2018.

[96] M. Yano. Discontinuous galerkin reduced basis empirical quadrature procedure for model reduction of parametrized nonlinear conservation laws. *Advances in Computational Mathematics*, 45:2287–2320, 2019.

[97] J. Zhang, B. O'Donoghue, and S. Boyd. Globally convergent type-I Anderson acceleration for nonsmooth fixed-point iterations. *SIAM Journal on Optimization*, 30(4):3170–3197, 2020.

[98] S. Zhang. A new family of stable mixed finite elements for the 3D Stokes equations. *Mathematics of Computation*, 74:543–554, 2005.