PHENOMENOLOGY AND COMPUTATIONS OF A REGULARIZATION OF THE NAVIER-STOKES EQUATIONS RELATED TO A NON-NEWTONIAN FLUID FLOW MODEL

SARA HRITZ
University of Pittsburgh
Department of Mathematics
saramhritz@gmail.com
2010

William Layton

PHENOMENOLOGY AND COMPUTATIONS OF A REGULARIZATION OF THE NAVIER-STOKES EQUATIONS RELATED TO A NON-NEWTONIAN FLUID FLOW MODEL

Sara Marie Hritz, MS

University of Pittsburgh, 2010

This paper analyzes the dynamics of non-Newtonian fluids, those whose viscosity is not constant. First, the Navier-Stokes equations are modified by introducing a new parameter with units of viscosity. Then, the energy equation and micro-scale of the model are derived. This allows the value of the parameter to be determined in order to make the micro-scale the order of the mesh width. Finally, the Finite Element Method with Backward Euler discretization is programmed using FreeFEM++ to simulate the model; a problem with known exact solution is used to test convergence of the method, and the step problem is also discussed.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1.0 THE PROBLEM

1.1 INTRODUCTION

Consider the flow of a fluid in a region Ω in $\mathbb{R}^3$. The Navier-Stokes equations (NSE) describe the motion of fluids. They are based on the conservation of mass and conservation of momentum laws. For Newtonian fluids, those with constant viscosity, that are incompressible or divergence free, the NSE are given by the following

$$\nabla \cdot u = 0 \tag{1.1.1}$$

$$u_t + u \cdot \nabla u - \nu \Delta u + \nabla p = f, in\ \Omega.$$

Here, *u(x, t)* is the fluid velocity vector, *v* is the constant kinematic viscosity, *p(x, t)* is the pressure, and *f(x, t)* is the body force vector. The NSE are nonlinear partial differential equations describing the velocity of a fluid as a function of space and time. The NSE will be accompanied by L-periodic boundary conditions on the domain $\Omega = (0, L)^3$, such that

$$u\left(x + L\widehat{e_j}\right) = u(x), \qquad j = 1, 2, 3$$

and u, u_o, p and f all have mean value zero,

$$\int_\Omega \varphi(x,t)dx = 0\,, \qquad for\ \varphi = u, u_o, p, and\ f.$$

The following non-Newtonian fluid flow model will be derived and analyzed:

$$\nabla \cdot u = 0$$

$$u_t - \alpha\tau\Delta u_t + u \cdot \nabla u - \nu\Delta u + \nabla p - \alpha\Delta(u - \bar{u}) = f,$$

where α is a positive constant parameter with units of viscosity and τ is a time scale with units of time. After taking the inner product of the NSE with u and then integrating over the domain Ω, the energy equation for this model becomes

$$\frac{d}{dt}\left(\int_{\Omega}\left(\frac{1}{2}|u|^2+\frac{\alpha\tau}{2}|\nabla u|^2\right)dx\right)+\int_{\Omega}(\nu|\nabla u|^2+\alpha\nabla(u-\bar{u}):\nabla u)dx=\int_{\Omega} f\cdot u dx\,.$$

This provides the kinetic energy and energy dissipation rate, allowing the energy cascade and micro-scale of the model to be found. In order to make the micro-scale the order of the mesh width, the parameter alpha should be chosen as

$$\alpha(\delta)=-\frac{1}{3}\sqrt[3]{\frac{1}{2}\left(-216\frac{U^3}{L}\delta^4+\sqrt{46{,}656\frac{U^6}{L^2}\delta^8-55{,}296\frac{U^9}{L^9}\delta^{12}\tau^3}\right)},$$

where δ is the length scale, τ is the time step, and U and L are the velocity and length at the large scales, respectively.

After analyzing the non-Newtonian fluid model, a program written in FreeFEM++ utilizes the Finite Element Method with Backward Euler time discretization to approximate the solution to the NSE. First, a problem is derived using an exact solution in order to test the first order convergence of the method that is expected. Then the solution to the step problem is computed, and the plots are discussed.

1.2 PHENOMENOLOGY OF THE NSE

The energy equation is derived by taking the inner product of (1.1.1) with u and then integrating over the domain Ω. The result is

$$\frac{d}{dt}\int_{\Omega}\frac{1}{2}|u|^2dx+\int_{\Omega}\nu|\nabla u|^2dx=\int_{\Omega} f\cdot u dx,$$

where $E(t)=\int_{\Omega}\frac{1}{2}|u|^2dx$ is the kinetic energy and $\varepsilon(t)=\int_{\Omega}\nu|\nabla u|^2dx$ is the energy dissipation rate. The energy cascade over the inertial range, which will be discussed in detail later, is given by

$$E(k)=c\varepsilon^{\frac{2}{3}}k^{-\frac{5}{3}},$$

Where *c* is the Kolmogorov constant, and *k* is the wave number. Two assumptions must be made in order to determine the micro-scale of the NSE. First, the Reynolds number *Re* at small scales, those with large wave numbers, is of order 1. Also, the energy in at large scales equals the energy out at small scales. Under these two assumptions, the bottom of the inertial range is at length scale η, where $\eta_{NSE} = O(Re^{-\frac{3}{4}})$ for 3*d* turbulence.

2.0 NON-NEWTONIAN FLUID FLOW MODEL

2.1 DERIVATION OF THE MODEL

Non-Newtonian fluids are those whose flow properties are not described by a single constant value of viscosity. Therefore, a change must be made to the NSE for these types of fluids. A proposed non-Newtonian fluid flow model is

$$\nabla \cdot u = 0 \tag{2.1.1}$$

$$u_t - \alpha\tau\Delta u_t + u \cdot \nabla u - \nu\Delta u + \nabla p = f,$$

where α is a positive parameter with units of viscosity and τ is a time scale with units of time.

The connection between this model and a numerical regularization arises from looking at the standard NSE implicit discretization:

$$\nabla \cdot u^{n+1} = 0$$

$$\frac{u^{n+1} - u^n}{\Delta t} + u^n \cdot \nabla u^{n+1} - \nu\Delta u^{n+1} + \nabla p^{n+1} = f^{n+1}.$$

If we rearrange and solve for the u^{n+1} terms, we get

$$\left[\frac{1}{\Delta t} I + u^n \cdot \nabla - \nu\Delta\right] u^{n+1} = \frac{u^n}{\Delta t} - \nabla p^{n+1} + f^{n+1},$$

where the spacial discretization has been suppressed.

Unfortunately, this system of equations is hard to solve if ν is small, so we increase "ν" by α on the left hand side (LHS) and adjust for this increase on the right hand side (RHS) as follows,

$$\left[\frac{1}{\Delta t} I + u^n \cdot \nabla - \nu\Delta - \alpha\Delta\right] u^{n+1} = \frac{u^n}{\Delta t} - \nabla p^{n+1} + f^{n+1} - \alpha\Delta u^n. \tag{2.1.2}$$

Now, if we go back to the NSE, we have

$$\frac{u^{n+1}-u^n}{\Delta t}-(\alpha\Delta t)\Delta\left(\frac{u^{n+1}-u^n}{\Delta t}\right)+u^n\cdot\nabla u^{n+1}-\nu\Delta u^{n+1}+\nabla p^{n+1}=f^{n+1}.$$

In the continuum limit, this gives (2.1.1) with $\tau = \Delta t$; however, it only causes a change in the kinetic energy and not the energy dissipation rate. Changing the model energy changes the details of the energy cascade through the inertial range but not the model micro-scale. To do this requires changing the energy dissipation rate. Therefore, a further modification must be made, leading to another model for non-Newtonian fluid flow.

In order to change the energy dissipation, we can take a spacial average of the velocity at the previous time step on the RHS of (2.1.2)

$$\left[\frac{1}{\Delta t}I+u^n\cdot\nabla-\nu\Delta-\alpha\Delta\right]u^{n+1}=\frac{u^n}{\Delta t}-\nabla p^{n+1}+f^{n+1}-\alpha\Delta\overline{u^n}. \tag{2.1.3}$$

This is precisely the method studied as a numerical approximation in [ALP]. We develop it herein as a continuum model. To do so, we rewrite the RHS of (2.1.3) as

$$\frac{u^n}{\Delta t}-\nabla p^{n+1}+f^{n+1}-\alpha\Delta u^n+\alpha\Delta(u^n-\overline{u^n}).$$

Thus, when we go back to the continuum limit, we have the following model

$$\nabla\cdot u=0 \tag{2.1.4}$$

$$u_t-\alpha\tau\Delta u_t+u\cdot\nabla u-\nu\Delta u+\nabla p-\alpha\Delta(u-\bar{u})=f,$$

with the hope that the new term $-\alpha\Delta(u-\bar{u})$ will cause a change in the energy dissipation and allow us to make the micro-scale the order of the local length scale. This paper will analyze (2.1.4), along with L-periodic boundary conditions.

3.0 THE ENERGY EQUATION

3.1 KINETIC ENERGY AND ENERGY DISSIPATION RATE

To find the energy equation, we take the inner product of (2.1.4) with *u* and integrate over the flow domain Ω. This gives the equation

$$\int_\Omega u_t \cdot u dx - \alpha\tau \int_\Omega \Delta u_t \cdot u dx + \int_\Omega u \cdot \nabla u \cdot u dx \tag{3.1.1}$$

$$-\nu \int_\Omega \Delta u \cdot u dx + \int_\Omega \nabla p \cdot u dx - \alpha \int_\Omega \Delta(u - \bar{u}) \cdot u dx = \int_\Omega f \cdot u dx.$$

Again, this will be accompanied by L-periodic boundary conditions on the domain $\Omega = (0, L)^3$, such that

$$u\left(x + L\hat{e}_j\right) = u(x), \qquad j = 1, 2, 3$$

and u, u_o, p and f all have mean value zero,

$$\int_\Omega v(x,t) dx = 0\,, \qquad for\ v = u, u_o, p, and\ f.$$

Calculations will be simplified using the summation convention that indices repeated in a multiplicative term are summed from 1 to 2, and indices occurring after a comma denote differentiation with respect to that variable, as in [Lay09]. For example, the vector product and tensor contraction can be written as:

$$u \cdot v = u_1 v_1 + u_2 v_2 = u_i v_i, and$$

$$T : S = \sum_{i=1}^{2} \sum_{j=1}^{2} T_{ij} S_{ij} = T_{ij} S_{ij}\,,$$

respectively. Also,

$$u_i u_i = |u|^2, u_{,j} = \frac{\partial u}{\partial x_j}, and\ u_{,jj} = \sum_{j=1}^{2} \frac{\partial^2 u}{\partial x_j^2}.$$

Using this notation, we can simplify the six terms on the LHS of (3.1.1) individually.

I. $\int_\Omega u_t \cdot u dx = \frac{d}{dt}\int_\Omega \frac{1}{2}|u|^2 dx$

II. $-\alpha\tau \int_\Omega \Delta u_t \cdot u dx = -\alpha\tau \int_\Omega \Delta \frac{d}{dt} u \cdot u dx = -\alpha\tau \frac{d}{dt}\int_\Omega \Delta u \cdot u dx = -\alpha\tau \frac{d}{dt}\int_\Omega u_{i,jj} u_i dx$

Using integration by parts (IBP), this is equivalent to

$$-\alpha\tau \frac{d}{dt}\int_\Omega [(u_{i,j}u_i)_{,j} - u_{i,j}u_{i,j}]dx. \tag{3.1.2}$$

By the Divergence Theorem, (3.1.2) equals

$$-\alpha\tau \frac{d}{dt}\int_{\partial\Omega} u_{i,j}u_i\hat{n}_j ds + \alpha\tau \frac{d}{dt}\int_\Omega u_{i,j}u_{i,j} dx,$$

where $\partial\Omega$ denotes the boundary of Ω, and $\hat{n}$ is the outward unit normal.

Because of the periodicity of the boundary, u is the same on each boundary, and the outward unit normals are of opposite signs. Therefore, $\int_{\partial\Omega} u_{i,j}u_i n_j ds = 0$. Hence,

$$-\alpha\tau \int_\Omega \Delta u_t \cdot u dx = \alpha\tau \frac{d}{dt}\int_\Omega u_{i,j}u_{i,j} dx = \alpha\tau \frac{d}{dt}\int_\Omega \frac{1}{2}|\nabla u|^2 dx = \frac{d}{dt}\int_\Omega \frac{\alpha\tau}{2}|\nabla u|^2 dx.$$

III. $\int_\Omega u \cdot \nabla u \cdot u dx = \int_\Omega u_i u_{j,i} u_j dx = \int_\Omega [(u_i u_j)_{,i} - u_{i,i}u_j]u_j dx$, by IBP. Since $u_{i,i} = \nabla \cdot u = 0$, then

$$\int_\Omega u \cdot \nabla u \cdot u dx = \int_\Omega [(u_i u_j u_j)_{,i} - u_i u_j u_{j,i}]dx, \tag{3.1.3}$$

again by IBP. By the Divergence Theorem and periodicity of the boundary, (3.1.3) equals

$$\int_{\partial\Omega} u_i u_j u_j \hat{n}_i ds - \int_\Omega u_i u_j u_{j,i} dx = -\int_\Omega u \cdot \nabla u \cdot u dx.$$

We have that $\int_\Omega u \cdot \nabla u \cdot u dx = -\int_\Omega u \cdot \nabla u \cdot u dx$, so $\int_\Omega u \cdot \nabla u \cdot u dx = 0$.

IV. $-\nu \int_\Omega \Delta u \cdot u dx = -\nu \int_\Omega u_{i,jj} u_i dx = -\nu \int_\Omega [(u_{i,j}u_i)_{,j} - u_{i,j}u_{i,j}]dx$

$$= -\nu \int_{\partial\Omega} u_{i,j}u_i\hat{n}_j ds + \nu \int_\Omega u_{i,j}u_{i,j} dx = \nu \int_\Omega |\nabla u|^2 dx,$$

by the same arguments used in *II*.

V. $\int_\Omega \nabla p \cdot u dx = \int_\Omega p_{,i} u_i dx = \int_\Omega [(u_i p)_{,i} - p u_{i,i}]\, dx,$ (3.1.4)

by IBP. By the Divergence Theorem and $\nabla \cdot u = 0$, (3.1.4) equals

$$\int_{\partial\Omega} u_i p \hat{n}_i ds.$$

Thus, this term is exactly zero due to the periodic boundary conditions.

VI. $-\alpha \int_\Omega \Delta(u-\bar{u}) \cdot u dx = -\alpha \int_\Omega (u-\bar{u})_{i,jj} u_i dx = -\alpha \int_\Omega [(u-\bar{u})_{i,j} u_i]_{,j} - (u-\bar{u})_{i,j} u_{i,j} dx =$ $-\alpha \int_{\partial\Omega} (u-\bar{u})_{i,j} u_i \hat{n}_i ds + \int_\Omega (u-\bar{u})_{i,j} u_{i,j} dx = \alpha \int_\Omega \nabla(u-\bar{u}) : \nabla u dx$

Hence, the energy equation becomes

$$\frac{d}{dt}\left(\int_\Omega \left(\frac{1}{2}|u|^2 + \frac{\alpha\tau}{2}|\nabla u|^2\right) dx\right) + \int_\Omega (\nu|\nabla u|^2 + \alpha\nabla(u-\bar{u}) : \nabla u) dx = \int_\Omega f \cdot u dx,$$

where $\int_\Omega \left(\frac{1}{2}|u|^2 + \frac{\alpha\tau}{2}|\nabla u|^2\right) dx = E_{model}$ and $\int_\Omega (\nu|\nabla u|^2 + \alpha\nabla(u-\bar{u}) : \nabla u) dx = \varepsilon_{model}$. Next, we need to prove that $\int_\Omega \alpha\nabla(u-\bar{u}) : \nabla u dx \geq 0$, to show that this term dissipates energy.

3.2 FOURIER SERIES ARGUMENT FOR ENERGY DISSIPATION RATE

In order to show that $\int_\Omega \alpha\nabla(u-\bar{u}) : \nabla u dx \geq 0$, Fourier series expansions will be used. The expansion for *u* is

$$u(x,t) = \sum_{\boldsymbol{k} \neq (0,0,0)} \hat{u}(\boldsymbol{k},t) e^{-i\boldsymbol{k}\cdot\boldsymbol{x}},$$

where $\boldsymbol{k} = \frac{2\pi}{L}\boldsymbol{n} = (k_1, k_2, k_3)$, $\boldsymbol{n} \in \mathbb{Z}^3$, $\boldsymbol{x} = (x_1, x_2, x_3)$, and the Fourier coefficient $\hat{u}(\boldsymbol{k},t) = \frac{1}{L^3}\int_\Omega u(x,t) e^{-i\boldsymbol{k}\cdot\boldsymbol{x}} d\boldsymbol{x}$. The wave number *k* is defined as $k = |\boldsymbol{k}|_2$. Next, we expand $\bar{u}$ by

$$\bar{u}(x,t) = \sum_{\boldsymbol{k} \neq (0,0,0)} b(\boldsymbol{k},t) e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}$$

and solve for the Fourier coefficient *b(**k**, t).* The differential filter in the periodic case is, given u, its average $\bar{u}$ is the solution of

$$-\delta^2\Delta\bar{u} + \bar{u} = u,$$

where δ is the length scale of the filter. We have

$$-\delta^2\Delta\bar{u} + \bar{u} = (-\delta^2\Delta + 1)\bar{u} = \sum_{\boldsymbol{k}\neq(0,0,0)} b(\boldsymbol{k},t)(-\delta^2\Delta + 1)e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}$$

$$= \sum_{\boldsymbol{k}\neq(0,0,0)} b(\boldsymbol{k},t)(\delta^2|\boldsymbol{k}|^2 + 1)e^{-i\boldsymbol{k}\cdot\boldsymbol{x}} = \sum_{\boldsymbol{k}\neq(0,0,0)} \hat{u}(\boldsymbol{k},t)e^{-i\boldsymbol{k}\cdot\boldsymbol{x}} = u\,,$$

and hence,

$$b(\boldsymbol{k},t)(-\delta^2\Delta + 1) = \hat{u}(\boldsymbol{k},t), i.e.$$

$$\bar{u}(x,t) = \sum_{\boldsymbol{k}\neq(0,0,0)} \frac{1}{\delta^2k^2 + 1}\hat{u}(\boldsymbol{k},t)e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}.$$

Now, we can compute $\nabla(u - \bar{u}) : \nabla u$ using these Fourier series expansions. First, we calculate $u - \bar{u}$,

$$u - \bar{u} = -\delta^2\Delta\bar{u} = \sum_{\boldsymbol{k}\neq(0,0,0)} \frac{\delta^2k^2}{\delta^2k^2 + 1}\hat{u}(\boldsymbol{k},t)e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}$$

and $\nabla(u - \bar{u})$,

$$\nabla(u - \bar{u}) = \sum_{\boldsymbol{k}} \frac{\delta^2k^2}{\delta^2k^2 + 1}\hat{u}(\boldsymbol{k}) \otimes (-i\boldsymbol{k})e^{-i\boldsymbol{k}\cdot\boldsymbol{x}},$$

and ∇u,

$$\nabla u = \sum_{\boldsymbol{k}} \hat{u}(\boldsymbol{k}) \otimes (-i\boldsymbol{k})e^{-i\boldsymbol{k}\cdot\boldsymbol{x}},$$

with $\otimes$ denoting the tensor product. Therefore,

$$(\nabla(u - \bar{u}), \nabla u) = (\sum_{\boldsymbol{k}} \frac{\delta^2k^2}{\delta^2k^2 + 1}\hat{u}(\boldsymbol{k}) \otimes (-i\boldsymbol{k})e^{-i\boldsymbol{k}\cdot\boldsymbol{x}}, \sum_{\boldsymbol{k}} \hat{u}(\boldsymbol{k}) \otimes (-i\boldsymbol{k})e^{-i\boldsymbol{k}\cdot\boldsymbol{x}})$$

$$= \sum_{\boldsymbol{k}} \frac{\delta^2k^2}{\delta^2k^2 + 1}\hat{u}(\boldsymbol{k}) \otimes (-i\boldsymbol{k})\,\overline{\hat{u}(\boldsymbol{k}) \otimes (-i\boldsymbol{k})},$$

by the property that the inner product $(f, g) = \sum_{\boldsymbol{k}} \hat{f}(\boldsymbol{k})\overline{\hat{g}(\boldsymbol{k})}$. So,

$$(\nabla(u - \bar{u}), \nabla u) = \sum_{\boldsymbol{k}} \frac{\delta^2k^2}{\delta^2k^2 + 1}|\hat{u}(\boldsymbol{k})|^2(-i)i|\boldsymbol{k}|^2 = \sum_{\boldsymbol{k}} \frac{\delta^2k^4}{\delta^2k^2 + 1}|\hat{u}(\boldsymbol{k})|^2 \geq 0,$$

thus showing that the new term dissipates energy.

4.0 PHENOMENOLOGY OF THE MODEL

4.1 THE ENERGY CASCADE

The energy cascade can be summarized in the following way, as in [P00]. Energy is input into the largest scales of the flow. There is an intermediate range, called the inertial range, in which nonlinearity drives the energy into smaller scales and conserves the global energy, since dissipation is negligible. At small enough scales, i.e. at the Kolmogorov micro-scale η, the energy dissipation becomes non negligible, and the energy is driven by the viscosity to zero exponentially fast. This theory is visualized in Figure 1.

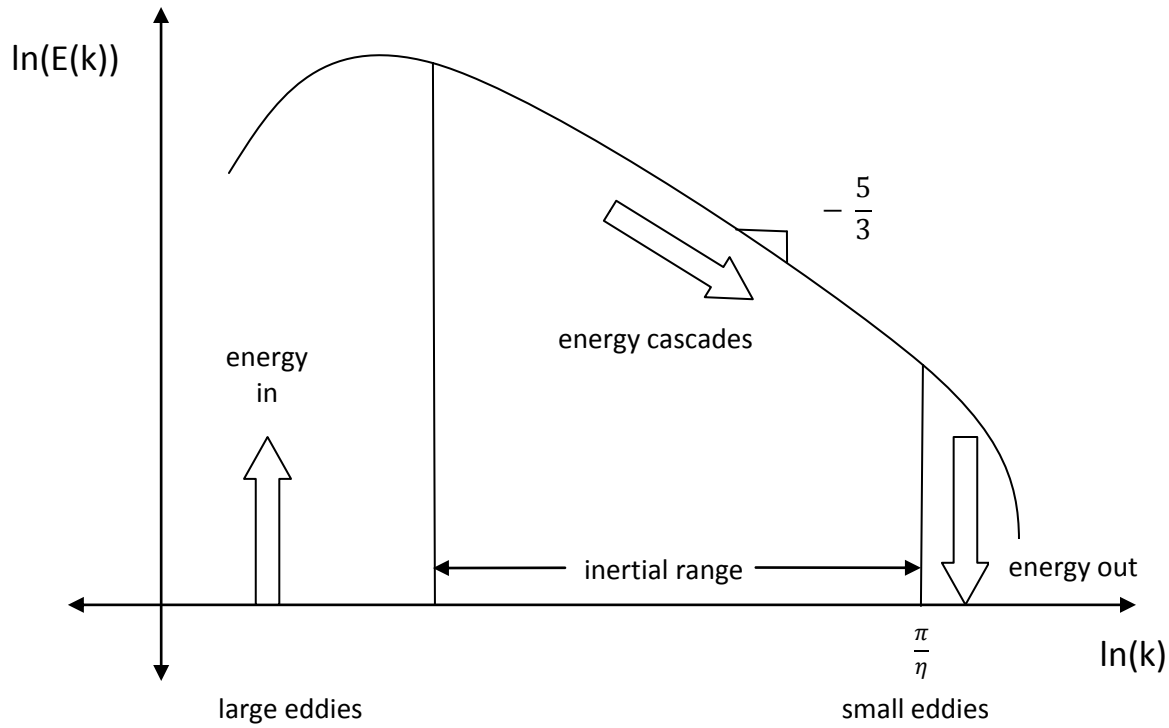

Figure 1. The Energy Cascade of the NSE

The K41 theory presented by Kolmogorov describing this energy cascade conjectures that the kinetic energy depends only on the energy dissipation rate ε and the wave number *k*. From this, we postulate the following relation over the inertial range

$$E(k) \simeq c\varepsilon^a k^b.$$

If this holds, then the units, denoted by [·], of the LHS must equal the units of the RHS. The units of the three quantities are

$$[k] = \frac{1}{length}, [\varepsilon] = \frac{length^2}{time^3}, and\ [E(k)] = \frac{length^3}{time^2}.$$

Inserting these units into the relation gives

$$\frac{length^3}{time^2} = \frac{length^{2a}}{time^{3a}} \frac{1}{length^b} = length^{2a-b} time^{-3a},$$

and thus $3 = 2a - b$ and $2 = 3a$, or $a = \frac{2}{3}$ and $b = -\frac{5}{3}$. Hence, Kolmogorov's Law follows

$$E(k) = c\varepsilon^{\frac{2}{3}}k^{-\frac{5}{3}},$$

over the inertial range, with *c* being the universal Kolmogorov constant whose value is generally believed to be between 1.4 and 1.7.

Next, we time average and expand $E_{model}(k)$. We have

$$\int_{\Omega} \left(\frac{1}{2}|u|^2 + \frac{\alpha\tau}{2}|\nabla u|^2\right) dx = \sum_{\boldsymbol{k}} \frac{1}{2}|\hat{u}(\boldsymbol{k})|^2 + \frac{\alpha\tau}{2}|\boldsymbol{k}|^2|\hat{u}(\boldsymbol{k})|^2$$

$$= \sum_{\boldsymbol{k}} (1 + \alpha\tau k^2)\frac{1}{2}|\hat{u}(\boldsymbol{k})|^2$$

$$= \sum_{k} (1 + \alpha\tau k^2) \sum_{k-1<|\boldsymbol{k}|\leq k} \frac{1}{2}|\hat{u}(\boldsymbol{k})|^2 = \sum_{k} (1 + \alpha\tau k^2)E(k).$$

We know that

$$E_{model}\,(k) = c_{model}\,\varepsilon_{model}^{2/3}\,k^{-5/3},$$

over the model's inertial range, so

$$E(k) = \frac{c_{model}\,\varepsilon_{model}^{2/3}\,k^{-5/3}}{1 + \alpha\tau k^2}.$$

Looking at two cases, if $1 > \alpha\tau k^2$, then

$$E(k) = c_{model}\,\varepsilon_{model}^{2/3}\,k^{-5/3},$$

and if $\alpha\tau k^2 > 1$, then

$$E(k) = c_{model}\,\varepsilon_{model}^{2/3}\,k^{-11/3}\alpha^{-1}\tau^{-1}.$$

Figure 2 illustrates the behavior of the energy cascade.

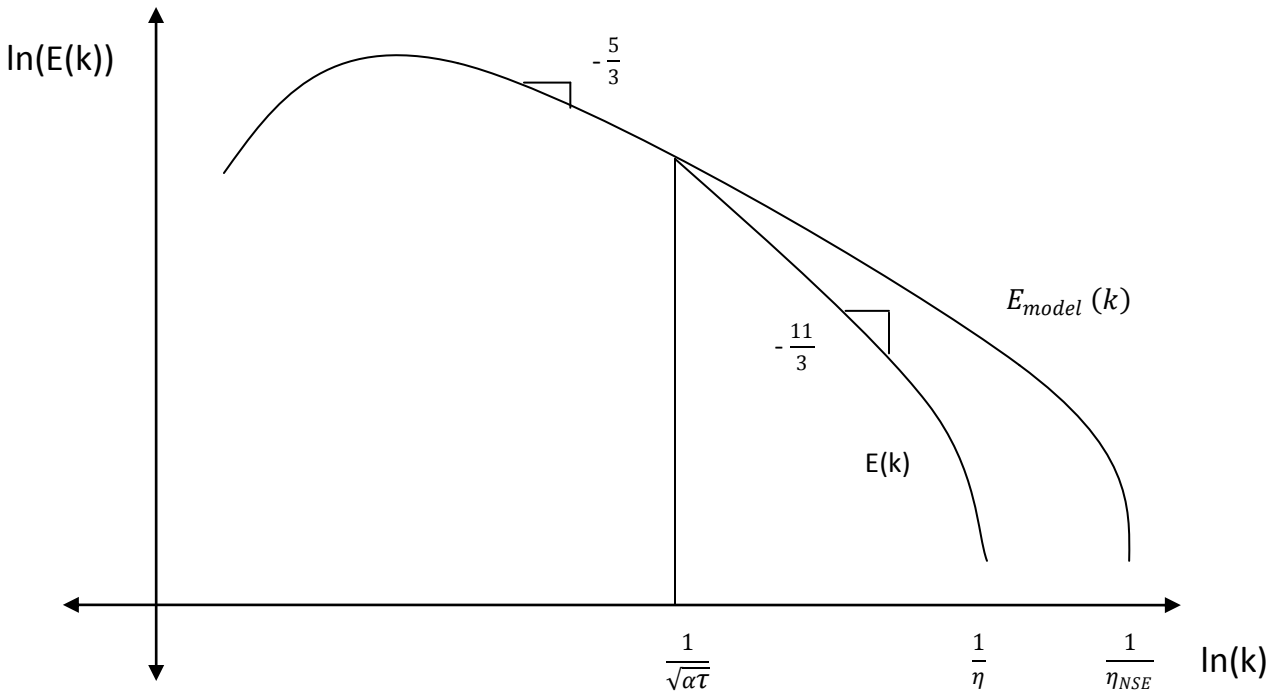

Figure 2. The Energy Cascade of the Non-Newtonian NSE

4.2 THE MICRO-SCALE

In order to determine the micro-scale of the model, we will make two assumptions about the flow, as in [L09]. Let the Reynolds number *Re* be defined by $\frac{uL}{\nu}$, a dimensionless parameter of the flow. The Reynolds number represents the ratio of the inertial forces to viscous forces of the flow. If *Re* is close to zero, then viscous forces dominate; for large *Re*, the viscous forces can be neglected. Let the reference large scale velocity and length be denoted by *U* and *L*, respectively. Also, at the scales of the smallest persistent eddies, i.e. the bottom of the inertial range, denote the smallest scale velocity and length by

u_{small} and η. The smallest scales Reynolds number measures the relative size of viscosity on the smallest persistent scales and is non negligible. Thus, we must have

$$Re_{small} \simeq 1,$$

where $Re \sim \frac{|nonlinear \quad terms|}{|viscous \quad terms|}$. This gives assumption 1 (A1):

$$1 = \frac{|u_{small} \cdot \nabla u_{small}|}{|\alpha\Delta(u_{small} - \bar{u}_{small})|} = \frac{\frac{1}{\eta}u_{small}^2}{\frac{\alpha}{\eta^2}\left(u_{small} - \left(\frac{\delta^2}{\eta^2}+1\right)^{-1} u_{small}\right)}$$

Next, there is an assumption of statistical equilibrium; energy in at large scales equals energy out at small scales. The large scale energy is $O(U^2)$, where $U = \frac{L}{T}$; therefore, the rate of energy in is $O\left(\frac{U^2}{T}\right) = O\left(\frac{U^3}{L}\right)$. Since $E_{model} = \int_\Omega \left(\frac{1}{2}|u|^2 + \frac{\alpha\tau}{2}|\nabla u|^2\right)dx$, which scales like $(1+\frac{\alpha\tau}{L^2})U^2$, the energy input rate is approximately $(1+\frac{\alpha\tau}{L^2})\frac{U^3}{L}$. The small scales energy dissipation from the viscous terms scales like

$$\varepsilon_{small} \simeq \nu|\nabla u_{small}|^2 + \alpha|\nabla(u_{small} - \bar{u}_{small}) : \nabla u_{small}|.$$

We will only be concerned for length scales that are much larger than η_{NSE}. Thus, ν will be negligible, and we have assumption 2 (A2):

$$\left(1+\frac{\alpha\tau}{L^2}\right)\frac{U^3}{L} = \frac{\alpha}{\eta^2}\left(u_{small} - \left(\frac{\delta^2}{\eta^2}+1\right)^{-1} u_{small}\right)u_{small}.$$

Solving A1 for u_{small}, we get

$$u_{small} = \frac{\alpha\delta^2}{\eta(\delta^2+\eta^2)}.$$

Plugging this in for u_{small} in A2 and solving for η, we have

$$\eta^4(\delta^2+\eta^2)^3 = \frac{\alpha^3\delta^6}{\left(1+\frac{\alpha\tau}{L^2}\right)\frac{U^3}{L}}.$$

Since we cannot solve explicitly for η, we will look at the case where $\delta = \eta$ ($= \eta_{model}$). We want to find how to pick α in order to make $\delta = \eta$, so we rearrange to get

$$\alpha^3 = \left(8\frac{U^3}{L}\delta^4\right)\left(1+\frac{\alpha\tau}{L^2}\right).$$

First, if $\alpha\tau << L^2$, then $\frac{\alpha\tau}{L^2}$ is negligible and we pick

$$\alpha = 2\frac{U}{\sqrt[3]{L}}\delta^{4/3}.$$

When $\frac{\alpha\tau}{L^2}$ becomes non negligible, then we have an equation of the form

$$\alpha^3 - a\alpha - b = 0, where\ a = 8\frac{U^3}{L^3}\tau\delta^4\ and\ b = 8\frac{U^3}{L}\delta^4. \tag{4.2.1}$$

We can now linearly approximate α as a function of a in order to get a nice equation for how to pick α:

$$\alpha(a) \cong \alpha(0) + \alpha^{'}(0)a$$

where

$$\alpha^{'}(a) = \frac{\alpha(a)}{3\alpha(a)^2 - a},$$

by implicit differentiation of (4.2.1). Plugging in $a = 0$, $\alpha(0) = 2\frac{U}{\sqrt[3]{L}}\delta^{\frac{4}{3}}$ and $\alpha^{'}(0) = \frac{1}{3}\frac{1}{\alpha(0)}$. Hence, we have the approximate formula for picking α to force $\eta_{model} = \delta$ in terms of a:

$$\alpha(a) \cong 2\frac{U}{\sqrt[3]{L}}\delta^{4/3} + \frac{1}{6}\frac{\sqrt[3]{L}}{U}\delta^{-4/3}a, with\ a = 8\frac{U^3}{L^3}\tau\delta^4.$$

Thus, after plugging in for a, we have an equation in terms of δ, after some simplification:

$$\alpha(\delta) \cong \frac{2U\delta^{4/3}}{L^{1/3}} + \frac{4U^2\tau\delta^{8/3}}{3L^{8/3}}.$$

We can find an exact equation for α by using the known formula for the roots of cubic functions. If we have a cubic of the form $x^3 + ax + b = 0$, then the real root is given by:

$$x = -\frac{1}{3}\sqrt[3]{\frac{1}{2}\left(27b + \sqrt{(27b)^2 - 4(-3a)^3}\right)}.$$

Plugging in $a = -8\frac{U^3}{L^3}\tau\delta^4\ and\ b = -8\frac{U^3}{L}\delta^4$, we get

$$\alpha(\delta) = -\frac{1}{3}\sqrt[3]{\frac{1}{2}\left(-216\frac{U^3}{L}\delta^4 + \sqrt{46{,}656\frac{U^6}{L^2}\delta^8 - 55{,}296\frac{U^9}{L^9}\delta^{12}\tau^3}\right)}.$$

5.0 FINITE ELEMENT METHOD WITH BACKWARD EULER

5.1 DERIVATION OF THE CODE

In order to use the Finite Element Method (FEM) to approximate the solution to the NSE in *2d* using FreeFEM++, we must first derive the weak or variational formulation. We have the following PDE, with nonhomogeneous Dirichlet boundary conditions:

$$u_t - \alpha\tau\Delta u_t + u \cdot \nabla u - \nu\Delta u + \nabla p - \alpha\Delta(u - \bar{u}) = f, in\ \Omega$$
$$\nabla \cdot u = 0, in\ \Omega$$
$$u|_{\partial\Omega} = g$$
$$u(x, 0) = u_0(x).$$

We want to find u ϵ $V = (H_0^1(\Omega))^2 = \{v \in (H^1(\Omega))^2 : v|_{\partial\Omega} = 0\}$ and p ϵ $P = L_0^2(\Omega) = \left\{p \in L^2(\Omega) : \int_\Omega pdx = 0\right\}$ such that

$$(u_t, v) - (\alpha\tau\Delta u_t, v) + (u \cdot \nabla u, v) - \nu(\Delta u, v) + (\nabla p, v) - \alpha(\Delta(u - \bar{u}), v) = (f, v), \forall\ v \in V$$
$$(\nabla \cdot u, q) = 0, \forall\ q \in P. \tag{5.1.1}$$

Applying IBP and the Divergence Theorem on the first equation of (5.1.1) leads to the variational formulation:

$$(u_t, v) + \alpha\tau(\nabla u_t, \nabla v) - \alpha\tau \int_{\partial\Omega} \nabla u_t \cdot v \cdot nds + b(u, u, v) + \nu(\nabla u, \nabla v) - \nu \int_{\partial\Omega} \nabla u \cdot v \cdot nds$$
$$-(p, \nabla \cdot v) + \int_{\partial\Omega} pv \cdot nds + \alpha(\nabla(u - \bar{u}), \nabla v) - \alpha \int_{\partial\Omega} \nabla(u - \bar{u}) \cdot v \cdot nds = (f, v), \forall\ v \in V$$

where $(\cdot,\cdot)$ denotes the L^2 inner product, n is the outward unit normal, and $b(a,b,c) \equiv \frac{1}{2}(a\cdot\nabla b, c) - \frac{1}{2}(a\cdot\nabla c, b)$ in order to make a linear system of equations. Discretizing in space using Backward Euler, we get:

$$\left(\frac{u^{n+1}-u^n}{\Delta t}, v\right) + \alpha\tau\left(\nabla\left(\frac{u^{n+1}-u^n}{\Delta t}\right), \nabla v\right) - \alpha\tau\int_{\partial\Omega}\nabla\left(\frac{u^{n+1}-u^n}{\Delta t}\right)\cdot v\cdot nds$$

$$+b(u^n, u^{n+1}, v) + \nu(\nabla u^{n+1}, \nabla v) - \nu\int_{\partial\Omega}\nabla u^{n+1}\cdot v\cdot nds - (p^{n+1}, \nabla\cdot v) + \int_{\partial\Omega}p^{n+1}v\cdot nds$$

$$+\alpha(\nabla(u^{n+1}-\overline{u^n}), \nabla v) - \alpha\int_{\partial\Omega}\nabla(u^{n+1}-\overline{u^n})\cdot v\cdot nds = (f^{n+1}, v), \forall\, v\in V^h$$

$$(\nabla\cdot u^{n+1}, q) = 0, \forall\, q\in P^h$$

where $t_n = n\Delta t$, $u^n(x) \cong u(t_n, x)$, $p^n(x) \cong p(t_n, x)$, V^h is the P2 velocity space of continuous piecewise quadratics, and P^h is the P1 pressure space of continuous piecewise linears.

5.2 CONVERGENCE TESTS

The FEM with Backward Euler discretization is a first order method. In order to test the convergence of the code, we will develop a problem from a known true solution. We consider the NSE

$$u_t + u\cdot\nabla u - \nu\Delta u + \nabla p = 0, in\ \Omega\subset\mathbb{R}^2$$

$$\nabla\cdot u = 0, in\ \Omega$$

with $u = a$ initially. Then, if a satisfies

$$\Delta a = -\lambda a,\ \nabla\cdot a = 0\ in\ \Omega, \tag{5.2.1}$$

then $u = e^{-\nu\lambda t}a$ satisfies the NSE with pressure p such that $\nabla p = -u\cdot\nabla u$ [W92]. If Ψ is an eigenfunction of the Laplacian, with eigenvalue λ, then $a = (\Psi_y, -\Psi_x)$ satisfies (5.2.1). Let $\Psi = \cos(x)\cos(y)$, and thus $\lambda = 1^2 + 1^2 = 2$. Then

$$u = e^{-\nu\lambda t}a = e^{-2\nu t}(-\cos(x)\sin(y), \sin(x)\cos(y)).$$

For the first problem, α, δ, and ν will be fixed in order to see what happens as the spacial step h and time step dt go to zero. The FreeFEM code is:

```
// Sara Hritz 3/10/2010
// Non-Newtonian Fluid Flow Model
// Solve u_t-alpha*tau*laplacian(u_t)+u.gradu-nu*laplacian(u)+gradp-alpha*laplacian(u-ubar)=f,
div(u)=0, in omega
// u=g on the boundary of omega
// uexact=exp(-nu*lambda*t)*a, where a=curl(psi)=(-cos(x)sin(y), sin(x)cos(y))
// psi=cos(x)cos(y), lambda=2
// ubar=(1/(lambda*delta^2+1))*u
// omega=(0,1)x(0,1)
// Fix alpha, delta, and nu; Take h and dt to 0

// time and parameter information
real dt = 1.0/10.0;          // time step size = tau
real TO = 0.0;               // initial time
real Tf = 1.0;               // final time
real t;
real nu = 1.0;               // viscosity
real alpha = (0.1)^(4/3);  // parameter
real delta = 0.1;            // differential filter length

// define macros
macro grad(u) [dx(u),dy(u)] //
macro div(u1,u2) (dx(u1)+dy(u2)) //
macro dot(u1,u2,v1,v2) (u1*v1+u2*v2) //
macro ugradv1(u1,u2,v1) (u1*dx(v1)+u2*dy(v1)) //
macro cc(u1,u2,v1,v2,w1,w2) (ugradv1(u1,u2,v1)*w1+ugradv1(u1,u2,v2)*w2) //
macro cch(u1,u2,v1,v2,w1,w2) (0.5*(cc(u1,u2,v1,v2,w1,w2)-cc(u1,u2,w1,w2,v1,v2))) //
macro contract(u1,u2,v1,v2) (dx(u1)*dx(v1)+dx(u2)*dx(v2)+dy(u1)*dy(v1)+dy(u2)*dy(v2)) //

// define the triangulated mesh
mesh Th=square(10,10);  // h=1/10
```

```
// define the called functions
func f1 = 0;  // RHS
func f2 = 0;
func g1 = exp(-2*nu*t)*(-cos(x)*sin(y));  // BC
func g2 = exp(-2*nu*t)*(sin(x)*cos(y));

// create the FE velocity space Vh of continuous piecewise quadratics and pressure space Ph of
continuous piecewise linears
fespace Vh(Th,P2);
fespace Ph(Th,P1);

// define the FE functions
Vh u1,u2,u1old,u2old,u1bar,u2bar,v1,v2;
Ph p,q;

// define the variational formulation of the NSE for Non-Newtonian fluids
problem NSE([u1,u2,p],[v1,v2,q]) =
  int2d(Th)(
    (1/dt)*dot(u1,u2,v1,v2)
    + 2*alpha*contract(u1,u2,v1,v2)
    + cch(u1old,u2old,u1,u2,v1,v2)
    + nu*contract(u1,u2,v1,v2)
    - p*div(v1,v2)
    + q*div(u1,u2))
  +int1d(Th)(
    -(2*alpha+nu)*cc(v1,v2,u1,u2,N.x,N.y)
    +p*dot(v1,v2,N.x,N.y))
  -int2d(Th)(
    (1/dt)*dot(u1old,u2old,v1,v2)
    + alpha*(contract(u1old,u2old,v1,v2)+contract(u1bar,u2bar,v1,v2))
    + dot(f1,f2,v1,v2))
  +int1d(Th)(
```

```
   alpha*(cc(v1,v2,u1old,u2old,N.x,N.y)+cc(v1,v2,u1bar,u2bar,N.x,N.y)))
  + on(1,2,3,4,u1=g1,u2=g2);

// define the variational formulation of the differential filter problem for u
problem dfiltu([u1bar,u2bar],[v1,v2]) =
  int2d(Th)(
   dot(u1bar,u2bar,v1,v2)
   + (delta^2)*contract(u1bar,u2bar,v1,v2))
  -int2d(Th)(
   u1*v1+u2*v2)
   + on(1,2,3,4,u1bar=(1/(2*(delta^2)+1))*g1,u2bar=(1/(2*(delta^2)+1))*g2);

// initialize variables before time stepping loop; u=a when t=0
u1 = -cos(x)*sin(y);
u2 = sin(x)*cos(y);
u1bar = (1/(2*(delta^2)+1))*u1;
u2bar = (1/(2*(delta^2)+1))*u2;
t = TO;

// begin time stepping loop
while (t < Tf)
{
u1old = u1;
u2old = u2;
t = t+dt;
// solve for [u1,u2,p]
NSE;
// solve for ubar at current time step
dfiltu;
plot([u1,u2],wait=0,value=true);
} // end while loop
```

```
// plot the computed pressure and velocity
plot(p,fill=1,wait=1,ps="pressurefinal1.eps",value=true);
plot([u1,u2],wait=1,ps="velocityfinal1.eps",value=true);

// compute the errors grad(pexact)-grad(p), (uexact-u)
Ph gradperr1 = abs(-ugradv1(u1,u2,u1)-dx(p));
Ph gradperr2 = abs(-ugradv1(u1,u2,u2)-dy(p));
Vh uerr1 = abs(g1-u1);
Vh uerr2 = abs(g2-u2);

// print the L2-norm of the errors in grad(p) and u
cout << "gradient of pressure error=" << sqrt(int2d(Th)(gradperr1^2+gradperr2^2)) << endl;
cout << "velocity error=" << sqrt(int2d(Th)(uerr1^2+uerr2^2)) << endl;
```

Running the code for different values of *h* and *dt*, starting with *h* = *dt* = $\frac{1}{10}$ and then cutting them in half, we get a table of the values of the L^2 norm of the errors in the gradient of the pressure and velocity values. We pick $\alpha = \delta^{\frac{4}{3}}$, $\delta = 0.1$, and $\nu = 1$. The errors are computed at the final time T = 1.0.

Table 1. Errors for Problem 1

h, dt	$\|\nabla p - \nabla p_h\|_{L^2(\Omega)}$	Ratio	Rate	$\|u - u_h\|_{L^2(\Omega)}$	Ratio	Rate
1/10	0.0196014			0.000132747		
1/20	0.011572	1.694	0.760	7.52336e-5	1.764	0.819
1/40	0.0055604	2.081	1.057	3.54803e-5	2.120	1.084
1/80	0.00253806	2.191	1.131	1.60419e-5	2.212	1.145

Both the pressure and velocity error ratios converge at the rate 2^p, with $p \cong 1$. This first order convergence is what we expected. As the mesh is refined, the computed solution of the model problem

converges slightly faster to the true solution of the NSE. The plots of the computed pressure and velocity for each value of *h* and *dt* at the final time are displayed in the following figures:

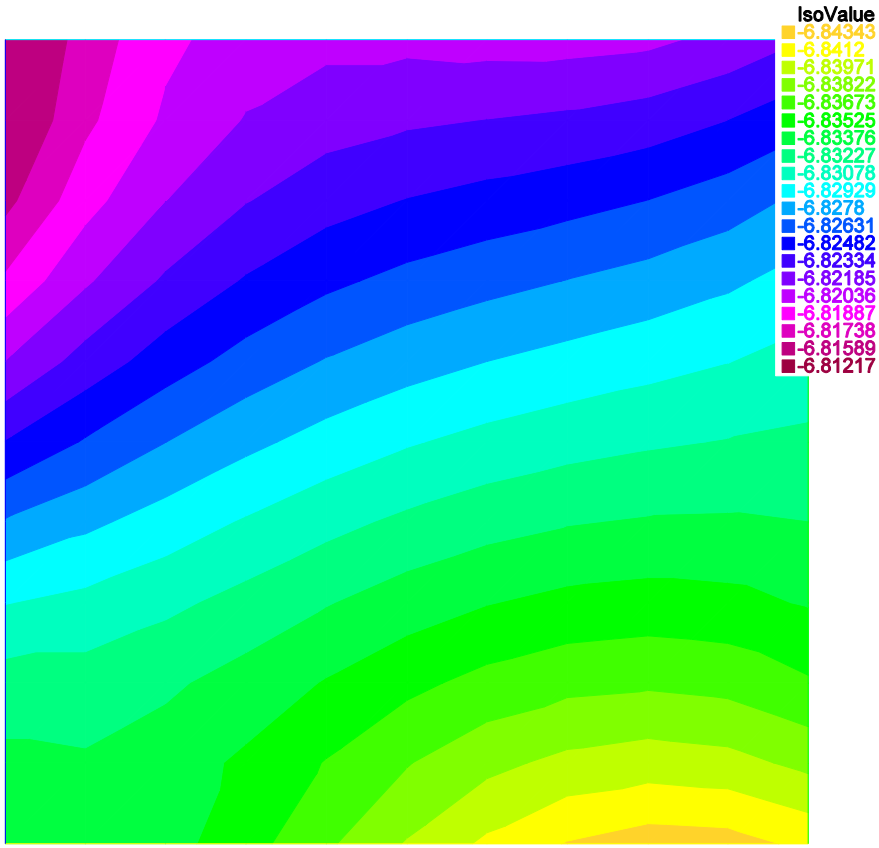

Figure 3. Problem 1, h = dt = 1/10, Computed Pressure

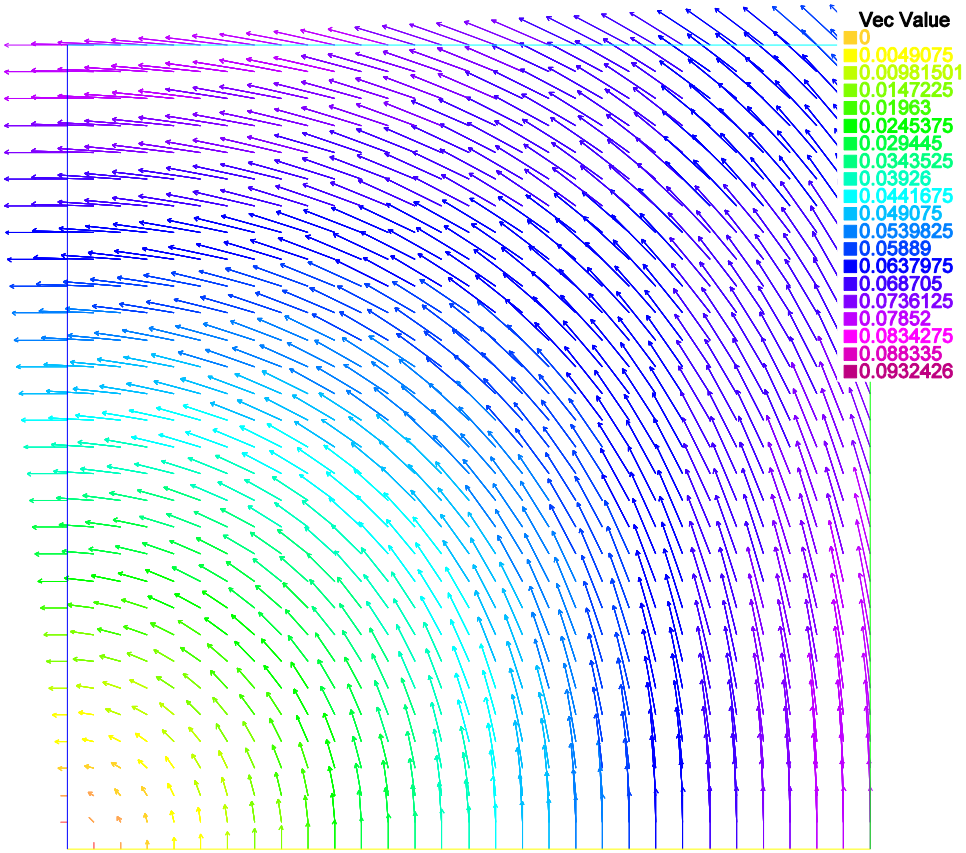

Figure 4. Problem 1, h = dt = 1/10, Computed Velocity

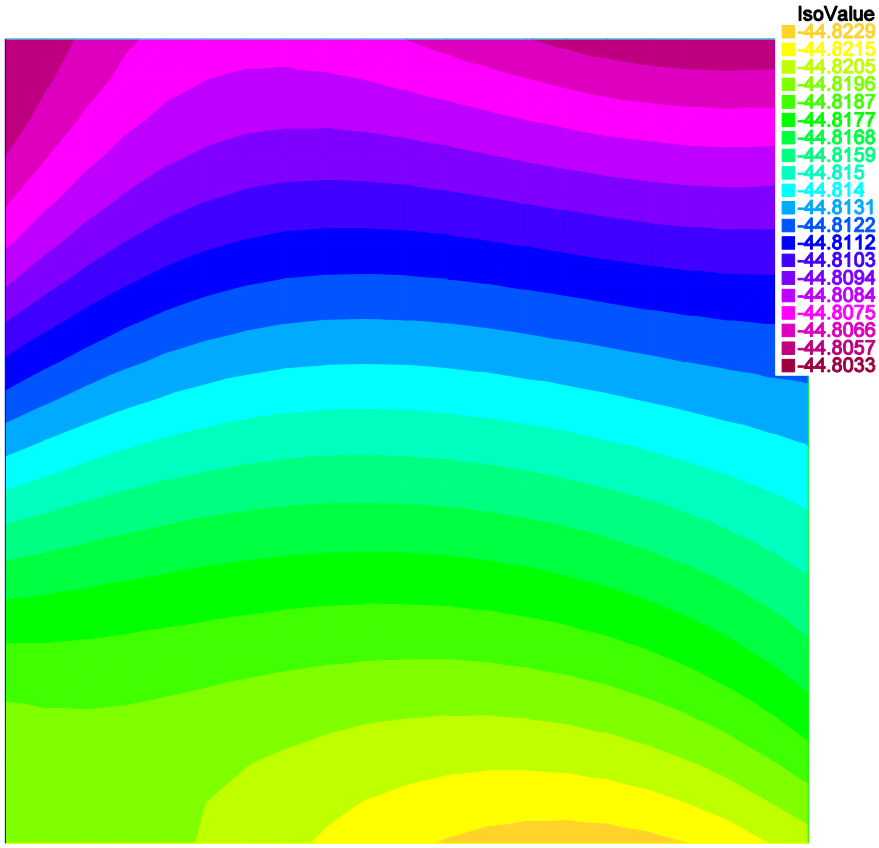

Figure 5. Problem 1, h = dt = 1/20, Computed Pressure

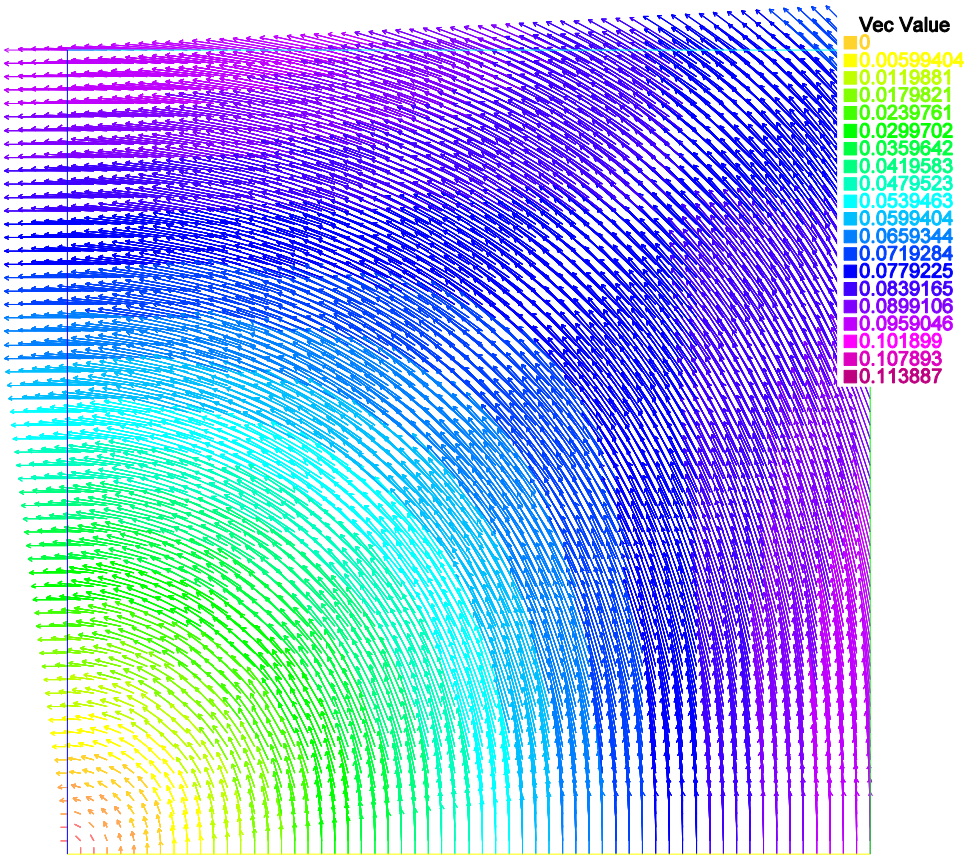

Figure 6. Problem 1, h = dt = 1/20, Computed Velocity

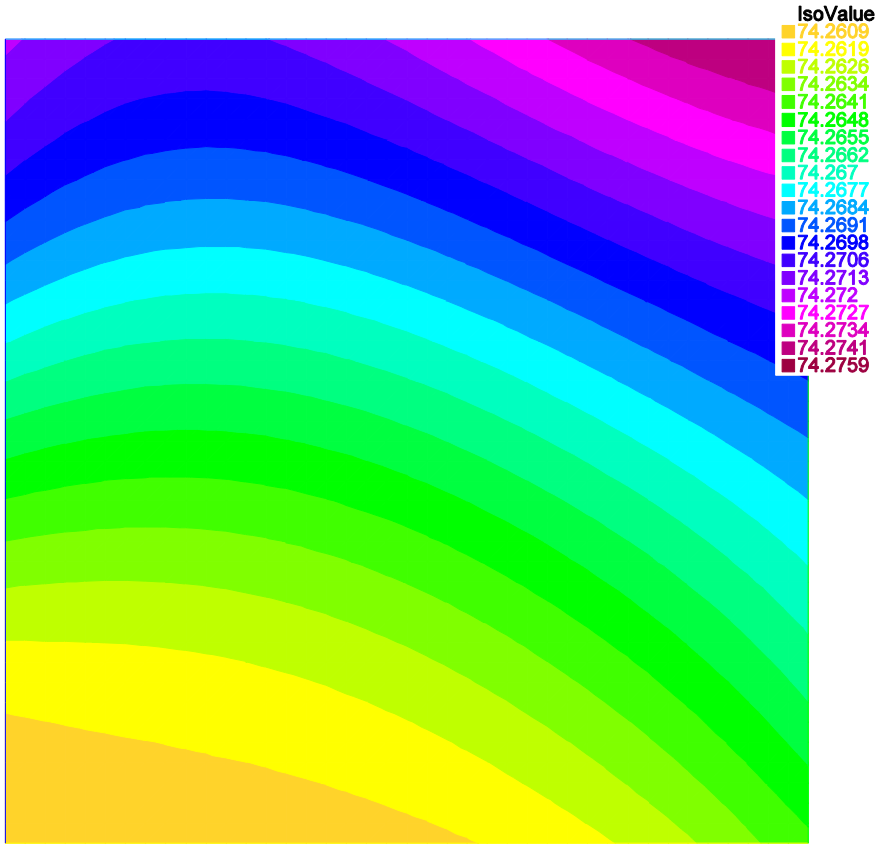

Figure 7. Problem 1, h = dt = 1/40, Computed Pressure

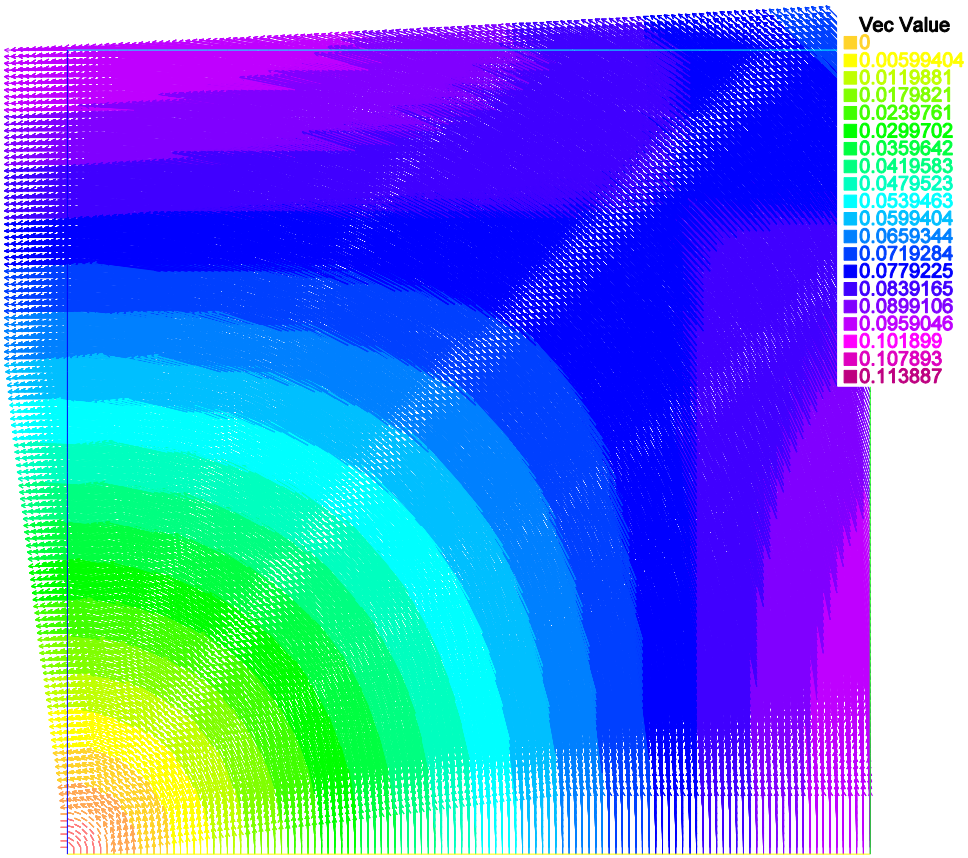

Figure 8. Problem 1, h = dt = 1/40, Computed Velocity

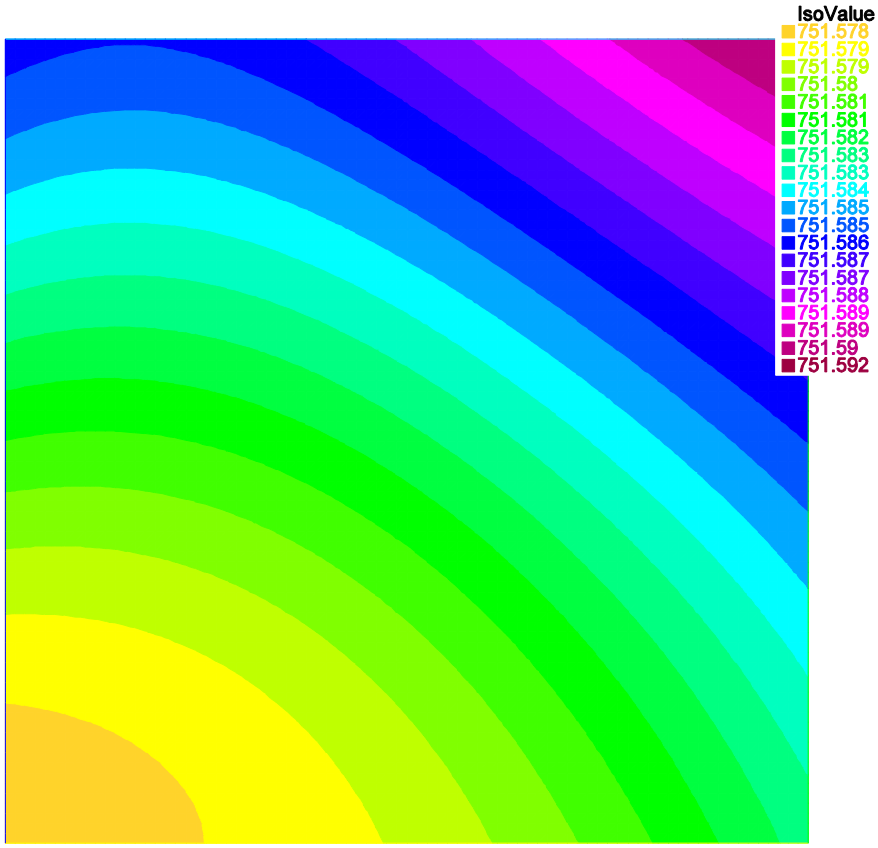

Figure 9. Problem 1, h = dt = 1/80, Computed Pressure

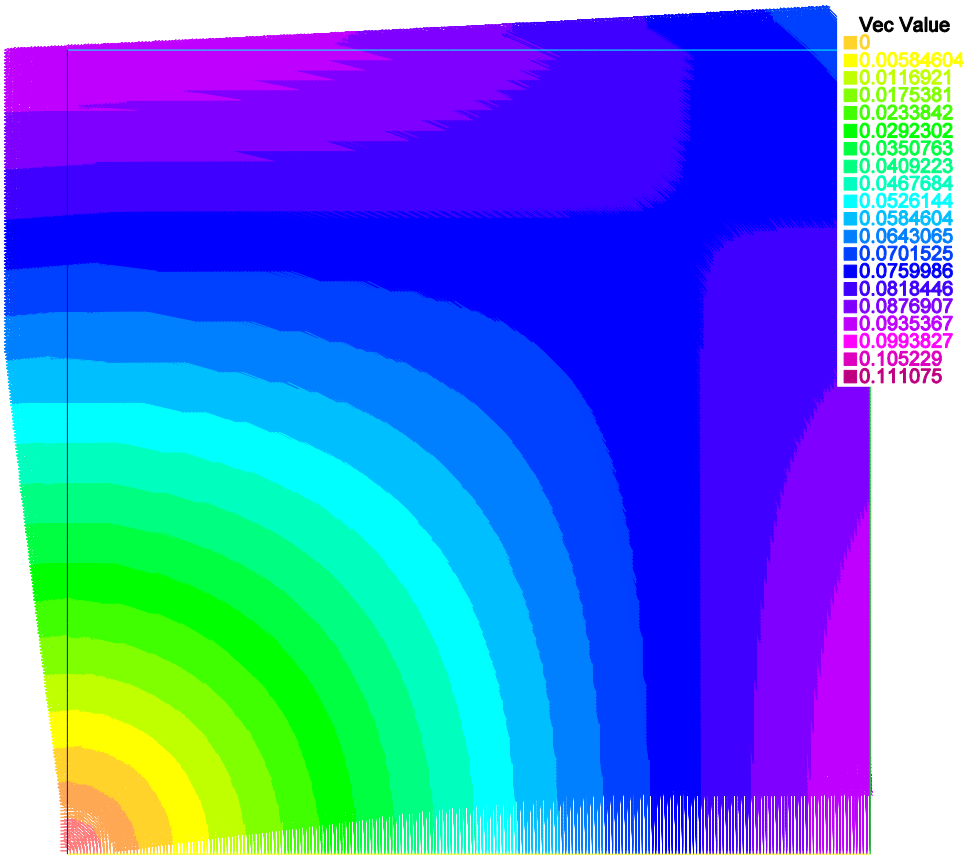

Figure 10. Problem 1, h = dt = 1/80, Computed Velocity

For the second problem, δ will no longer be fixed and will depend on *h*, and we will see what happens as *h* and *dt* go to zero. The code for this problem is:

```
// Sara Hritz 3/10/2010
// Non-Newtonian Fluid Flow Model
// Solve u_t-alpha*tau*laplacian(u_t)+u.gradu-nu*laplacian(u)+gradp-alpha*laplacian(u-ubar)=f,
div(u)=0, in omega
// u=g on the boundary of omega
// uexact=exp(-nu*lambda*t)*a, where a=curl(psi)=(-cos(x)sin(y), sin(x)cos(y))
```

```
// psi=cos(x)cos(y), lambda=2
// ubar=(1/(lambda*delta^2+1))*u
// omega=(0,1)x(0,1)
// Fix nu; Take delta=3h and h, dt to 0

// time and parameter information
real h = 1.0/10.0;              // spacial step
real dt = h;                    // time step size = tau
real TO = 0.0;                  // initial time
real Tf = 1.0;                  // final time
real t;
real nu = 1.0;                  // viscosity
real delta = 3*h;               // differential filter length
real alpha = (delta)^(4/3);  // parameter

// define macros
macro grad(u) [dx(u),dy(u)] //
macro div(u1,u2) (dx(u1)+dy(u2)) //
macro dot(u1,u2,v1,v2) (u1*v1+u2*v2) //
macro ugradv1(u1,u2,v1) (u1*dx(v1)+u2*dy(v1)) //
macro cc(u1,u2,v1,v2,w1,w2) (ugradv1(u1,u2,v1)*w1+ugradv1(u1,u2,v2)*w2) //
macro cch(u1,u2,v1,v2,w1,w2) (0.5*(cc(u1,u2,v1,v2,w1,w2)-cc(u1,u2,w1,w2,v1,v2))) //
macro contract(u1,u2,v1,v2) (dx(u1)*dx(v1)+dx(u2)*dx(v2)+dy(u1)*dy(v1)+dy(u2)*dy(v2)) //

// define the triangulated mesh
mesh Th=square(1/h,1/h);

// define the called functions
func f1 = 0;  // RHS
func f2 = 0;
func g1 = exp(-2*nu*t)*(-cos(x)*sin(y));  // BC
func g2 = exp(-2*nu*t)*(sin(x)*cos(y));
```

```
// create the FE velocity space Vh of continuous piecewise quadratics and pressure space Ph of
continuous piecewise linears
fespace Vh(Th,P2);
fespace Ph(Th,P1);

// define the FE functions
Vh u1,u2,u1old,u2old,u1bar,u2bar,v1,v2;
Ph p,q;

// define the variational formulation of the NSE for Non-Newtonian fluids
problem NSE([u1,u2,p],[v1,v2,q]) =
  int2d(Th)(
    (1/dt)*dot(u1,u2,v1,v2)
    + 2*alpha*contract(u1,u2,v1,v2)
    + cch(u1old,u2old,u1,u2,v1,v2)
    + nu*contract(u1,u2,v1,v2)
    - p*div(v1,v2)
    + q*div(u1,u2))
  +int1d(Th)(
    -(2*alpha+nu)*cc(v1,v2,u1,u2,N.x,N.y)
    +p*dot(v1,v2,N.x,N.y))
  -int2d(Th)(
    (1/dt)*dot(u1old,u2old,v1,v2)
    + alpha*(contract(u1old,u2old,v1,v2)+contract(u1bar,u2bar,v1,v2))
    + dot(f1,f2,v1,v2))
  +int1d(Th)(
    alpha*(cc(v1,v2,u1old,u2old,N.x,N.y)+cc(v1,v2,u1bar,u2bar,N.x,N.y)))
  + on(1,2,3,4,u1=g1,u2=g2);

// define the variational formulation of the differential filter problem for u
problem dfiltu([u1bar,u2bar],[v1,v2]) =
  int2d(Th)(
```

```
   dot(u1bar,u2bar,v1,v2)
   + (delta^2)*contract(u1bar,u2bar,v1,v2))
  -int2d(Th)(
   u1*v1+u2*v2)
   + on(1,2,3,4,u1bar=(1/(2*(delta^2)+1))*g1,u2bar=(1/(2*(delta^2)+1))*g2);

// initialize variables before time stepping loop
u1 = -cos(x)*sin(y);
u2 = sin(x)*cos(y);
u1bar = (1/(2*(delta^2)+1))*u1;
u2bar = (1/(2*(delta^2)+1))*u2;
t = TO;

// begin time stepping loop
while (t < Tf)
{
u1old = u1;
u2old = u2;
t = t+dt;
// solve for [u1,u2,p]
NSE;
// solve for ubar at current time step
dfiltu;
plot([u1,u2],wait=0,value=true);
} // end while loop

// plot the computed pressure and velocity
plot(p,fill=1,wait=1,ps="pressurefinal1.eps",value=true);
plot([u1,u2],wait=1,ps="velocityfinal1.eps",value=true);

// compute the errors grad(pexact)-grad(p), (uexact-u)
Ph gradperr1 = abs(-ugradv1(u1,u2,u1)-dx(p));
```

```
Ph gradperr2 = abs(-ugradv1(u1,u2,u2)-dy(p));
Vh uerr1 = abs(g1-u1);
Vh uerr2 = abs(g2-u2);

// print the L2-norm of the errors in grad(p) and u
cout << "gradient of pressure error=" << sqrt(int2d(Th)(gradperr1^2+gradperr2^2)) << endl;
cout << "velocity error=" << sqrt(int2d(Th)(uerr1^2+uerr2^2)) << endl;
```

As we decrease the value of *h* and *dt*, we get a table of errors. We pick $\alpha = \delta^{\frac{4}{3}}$, $\delta = 3h$, and $\nu = 1$.

Table 2. Errors for Problem 2

h, dt	$\|\nabla p - \nabla p_h\|_{L^2(\Omega)}$	Error Ratio	Rate	$\|u - u_h\|_{L^2(\Omega)}$	Error Ratio	Rate
1/10	0.0241666			0.000139179		
1/20	0.0124747	1.937	0.954	7.78572e-5	1.788	0.838
1/40	0.00533242	2.339	1.226	3.45882e-5	2.251	1.171
1/80	0.00233252	2.286	1.193	1.52253e-5	2.272	1.184

Again, we see the errors converging at a rate of about 1, giving first order convergence, and the convergence increases slightly as the mesh is refined. We look at the plots for different values of *h* and *dt*.

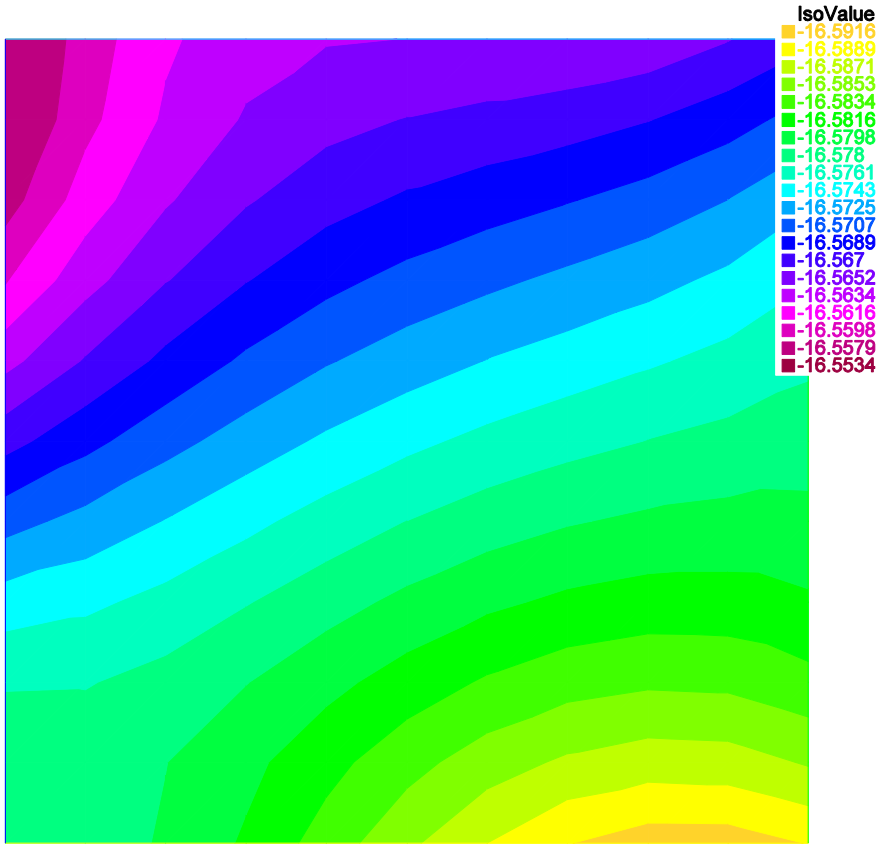

Figure 11. Problem 2, h = dt = 1/10, Computed Pressure

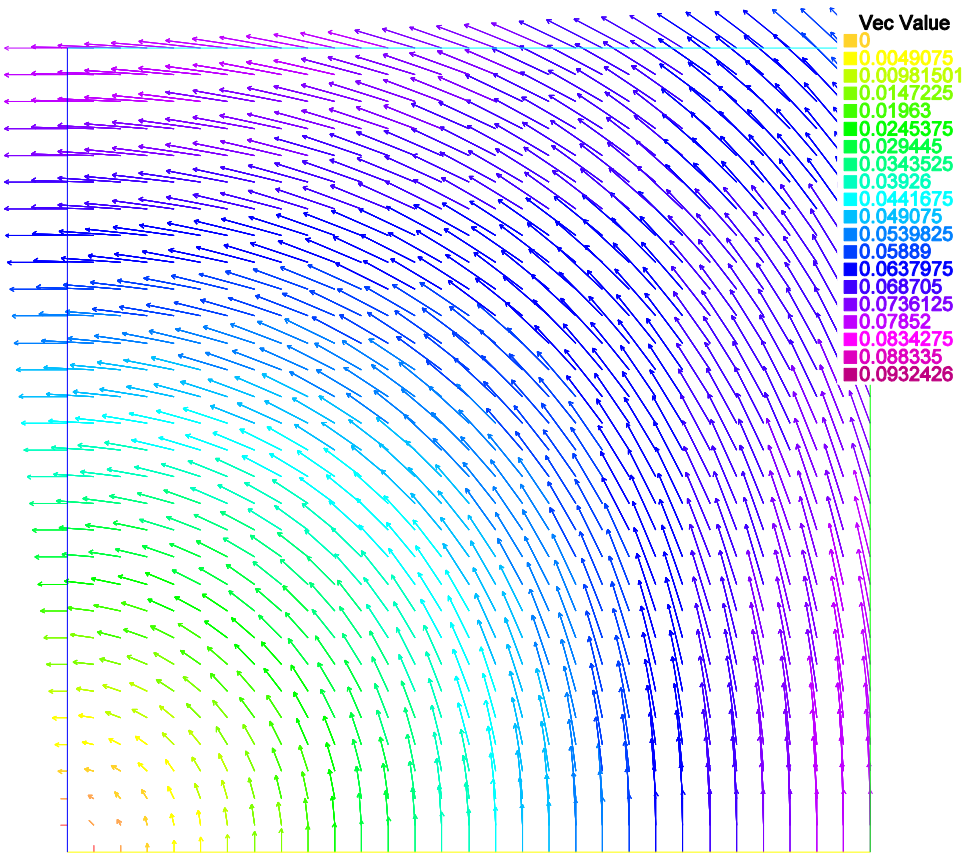

Figure 12. Problem 2, h = dt = 1/10, Computed Velocity

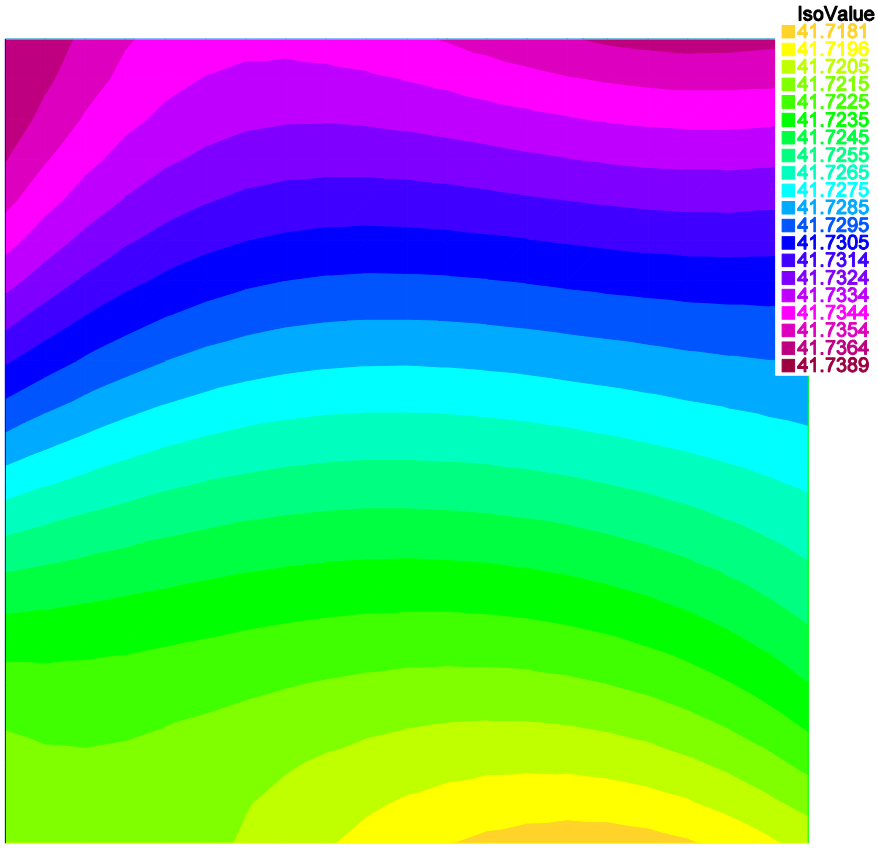

Figure 13. Problem 2, h = dt = 1/20, Computed Pressure

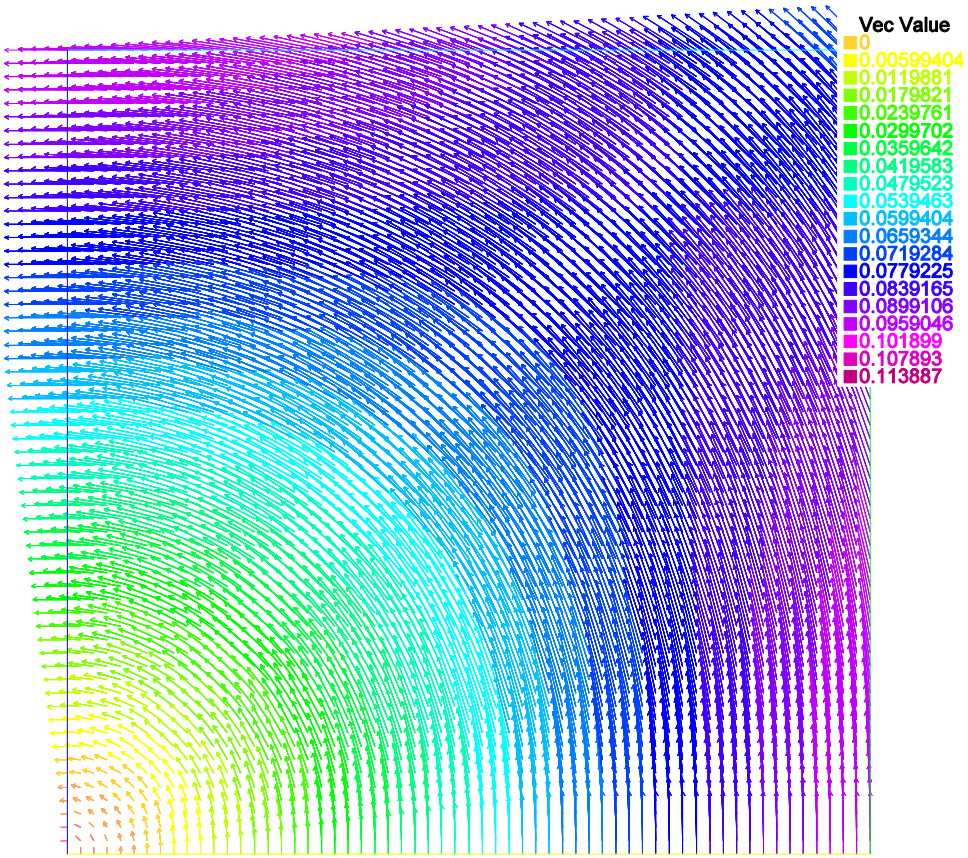

Figure 14. Problem 2, h = dt = 1/20, Computed Velocity

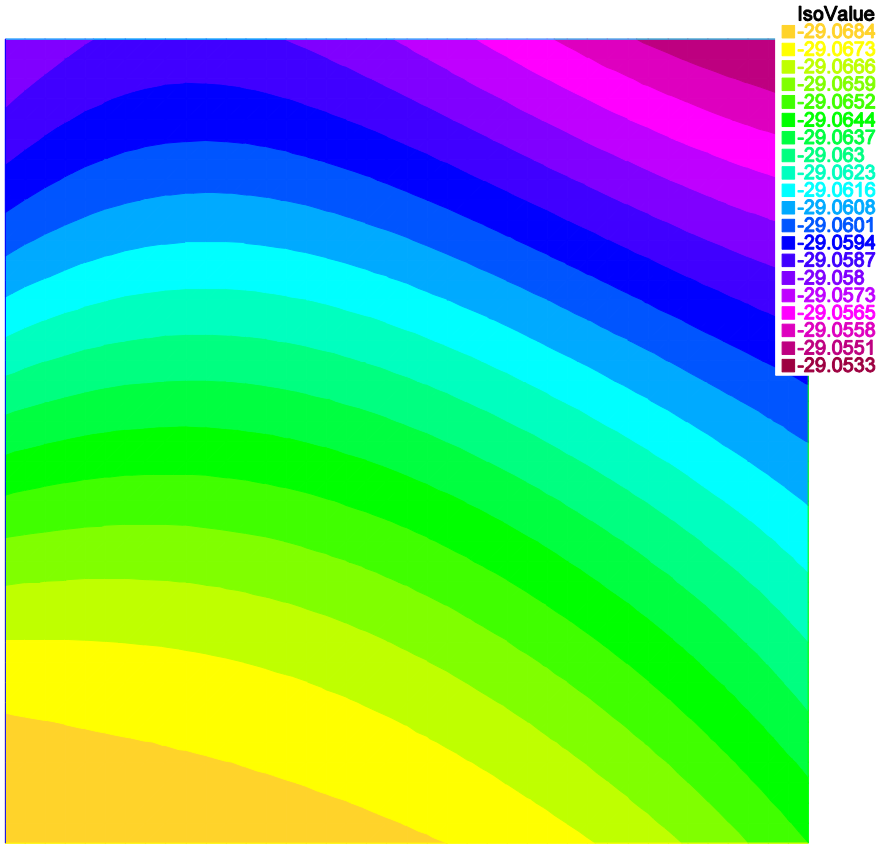

Figure 15. Problem 2, h = dt = 1/40, Computed Pressure

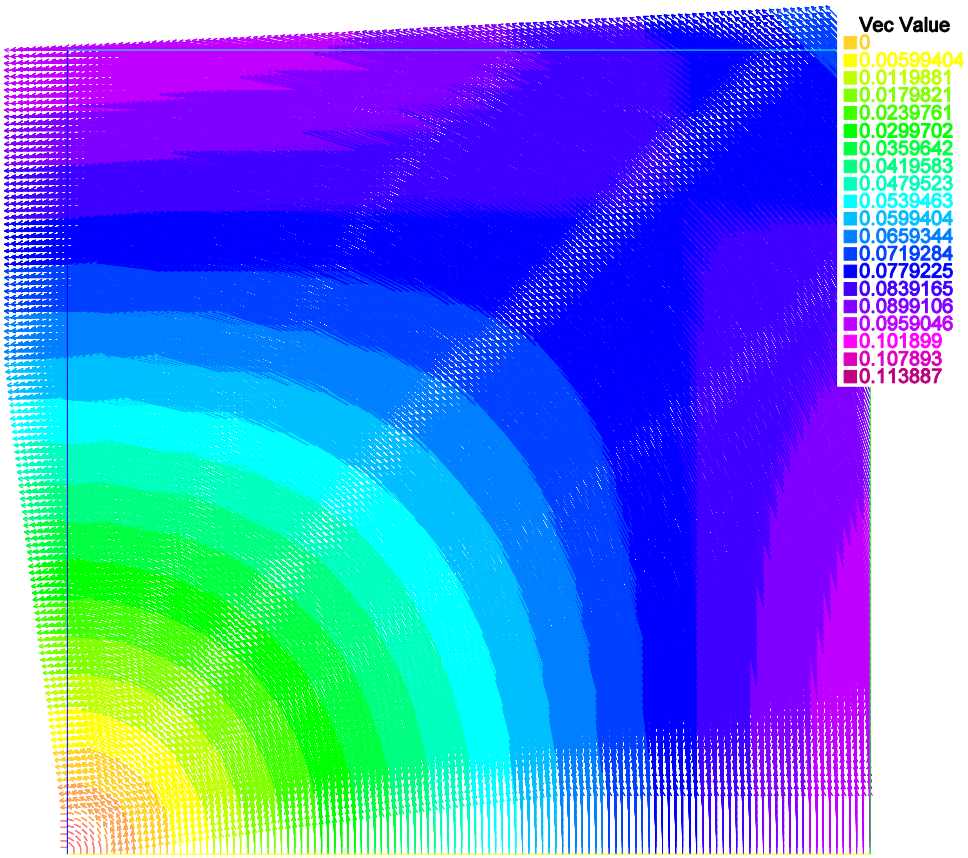

Figure 16. Problem 2, h = dt = 1/40, Computed Velocity

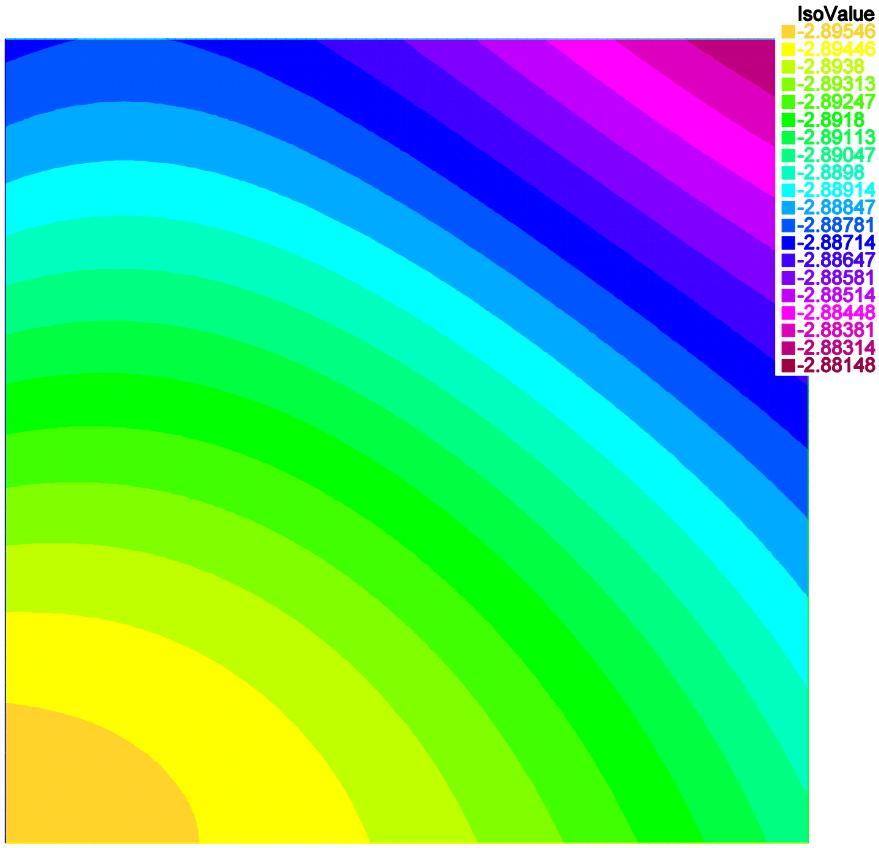

Figure 17. Problem 2, h = dt = 1/80, Computed Pressure

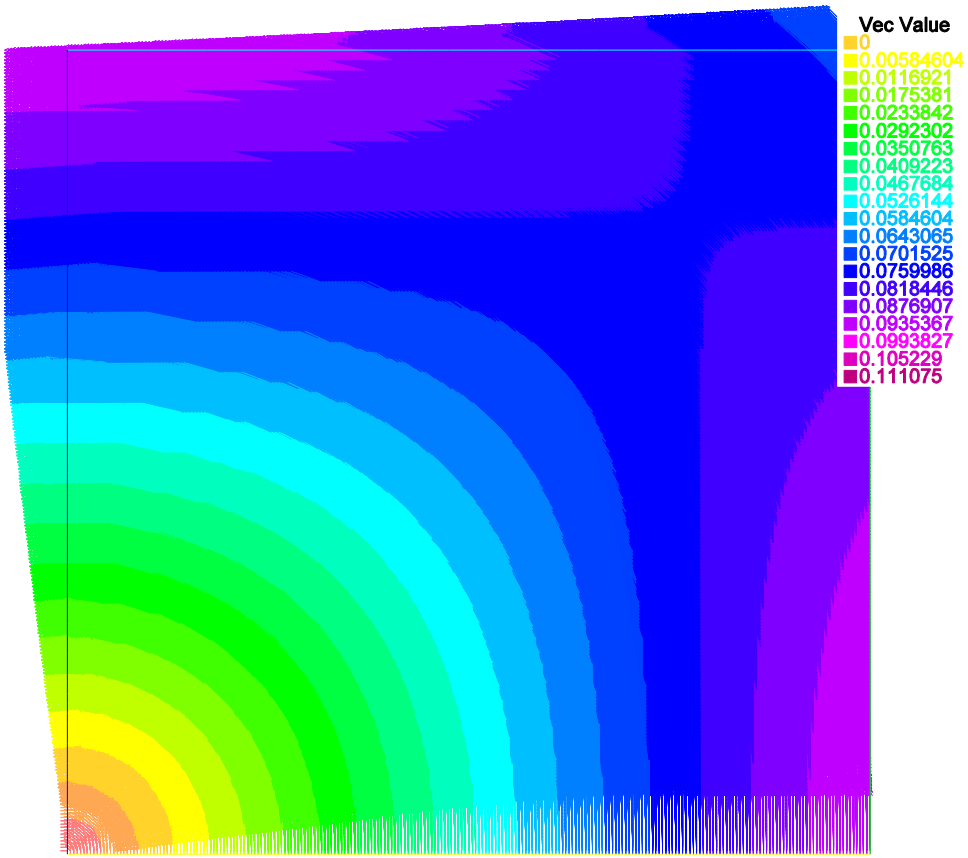

Figure 18. Problem 2, h = dt = 1/80, Computed Velocity

5.3 THE STEP PROBLEM

In order to see how the non-Newtonian fluid model handles a more realistic problem, we will look at the step problem, which has a more complex domain. The domain will be reduced by taking $x \in [0, 20]$. We will take a parabolic boundary condition on the left boundary for input to the flow, homogeneous Dirichlet boundary conditions on the top and bottom, and a "do nothing" boundary condition on the right. We will examine the plots of the computed pressure and velocity as the final time is increased, for

fixed values of α, δ, dt, and ν. We pick $\alpha = \delta^{\frac{4}{3}}$, $\delta = 0.05(h_{max} + h_{min})$, dt = 0.1 and $\nu = \frac{1}{600}$. The code for the step problem is:

```
// Sara Hritz 3/10/2010
// Non-Newtonian Fluid Flow Model
// Solve u_t-alpha*tau*laplacian(u_t)+u.gradu-nu*laplacian(u)+gradp-alpha*laplacian(u-ubar)=f,
div(u)=0, in omega
// Solve the step problem on [0,20]x[0,10]

// time and parameter information
real dt = 0.1;            // time step size = tau
real TO = 0.0;            // initial time
real Tf = 10.0;           // final time
real nu = 1.0/600.0;  // viscosity
real alpha;               // parameter
real delta;               // differential filter length

// define macros
macro grad(u) [dx(u),dy(u)] //
macro div(u1,u2) (dx(u1)+dy(u2)) //
macro dot(u1,u2,v1,v2) (u1*v1+u2*v2) //
macro ugradv1(u1,u2,v1) (u1*dx(v1)+u2*dy(v1)) //
macro cc(u1,u2,v1,v2,w1,w2) (ugradv1(u1,u2,v1)*w1+ugradv1(u1,u2,v2)*w2) //
macro cch(u1,u2,v1,v2,w1,w2) (0.5*(cc(u1,u2,v1,v2,w1,w2)-cc(u1,u2,w1,w2,v1,v2))) //
macro contract(u1,u2,v1,v2) (dx(u1)*dx(v1)+dx(u2)*dx(v2)+dy(u1)*dy(v1)+dy(u2)*dy(v2)) //

// define the triangulated mesh
border A1(t=0,5){x=t; y=0; label=1;};
border A2(t=0,3){x=6+t; y=0; label=2;};
border A3(t=0,11){x=9+t; y=0; label=9;};
border S1(t=0,1){x=5; y=t; label=3;};
border S2(t=0,1){x=5+t; y=1; label=4;};
```

```
border S3(t=0,1){x=6; y=1-t; label=5;};
border B(t=0,10){x=20; y=t; label=6;};
border C(t=0,20){x=20-t; y=10; label=7;};
border D(t=0,10){x=0; y=10-t; label=8;};
mesh Th=buildmesh(A1(4)+A2(12)+A3(20)+S1(2)+S2(2)+S3(4)+B(24)+C(32)+D(24));

// define the called functions
func f1 = 0;          // RHS
func f2 = 0;
func g = y*(10-y)/25;  // left BC

// create the FE velocity space Vh of continuous piecewise quadratics and pressure space Ph of
continuous piecewise linears
fespace Vh(Th,P2);
fespace Ph(Th,P1);

// define the FE functions
Vh u1,u2,u1old,u2old,u1bar,u2bar,v1,v2;
Vh h=hTriangle;
Ph p,q;

// define the variational formulation of the NSE for Non-Newtonian fluids
problem NSE([u1,u2,p],[v1,v2,q]) =
  int2d(Th)(
    (1/dt)*dot(u1,u2,v1,v2)
    + 2*alpha*contract(u1,u2,v1,v2)
    + cch(u1old,u2old,u1,u2,v1,v2)
    + nu*contract(u1,u2,v1,v2)
    - p*div(v1,v2)
    + q*div(u1,u2))
  +int1d(Th,6)(
    -2*alpha*cc(v1,v2,u1,u2,N.x,N.y))
```

```
 -int2d(Th)(
   (1/dt)*dot(u1old,u2old,v1,v2)
   + alpha*(contract(u1old,u2old,v1,v2)+contract(u1bar,u2bar,v1,v2))
   + dot(f1,f2,v1,v2))
 +int1d(Th,6)(
   alpha*(cc(v1,v2,u1old,u2old,N.x,N.y)+cc(v1,v2,u1bar,u2bar,N.x,N.y)))
 + on(1,2,3,4,5,7,9,u1=0,u2=0)
 + on(8,u1=g,u2=0);

// define the variational formulation of the differential filter problem for u
problem dfiltu([u1bar,u2bar],[v1,v2]) =
 int2d(Th)(
   dot(u1bar,u2bar,v1,v2)
   + (delta^2)*contract(u1bar,u2bar,v1,v2))
 -int2d(Th)(
   u1*v1+u2*v2)
   + on(1,2,3,4,5,7,9,u1bar=0,u2bar=0)
   + on(8,u1bar=g,u2bar=0);

// initialize variables before time stepping loop
u1 = 0.0;
u2 = 0.0;
u1bar = 0.0;
u2bar = 0.0;
real t = TO;
delta = 0.05*(h[].max + h[].min);  //best for coarse, non-uniform meshes
alpha = delta^(4/3);

// begin time stepping loop
while (t < Tf)
{
u1old = u1;
```

```
u2old = u2;
t = t+dt;
// solve for [u1,u2,p]
NSE;
// solve for ubar at current time step
dfiltu;
plot([u1,u2],wait=0,value=true);
} // end while loop

// plot the computed pressure and velocity
plot(p,fill=1,wait=1,ps="pressure1.eps",value=true);
plot([u1,u2],wait=1,ps="velocityfinal1.eps",value=true);
```

We obtain the plots of the computed pressure and velocity at the final times T = 10, T = 20, and T = 40. A zoom of the velocity field after the step is also shown.

Figure 19. Step Problem, T = 10, Computed Pressure

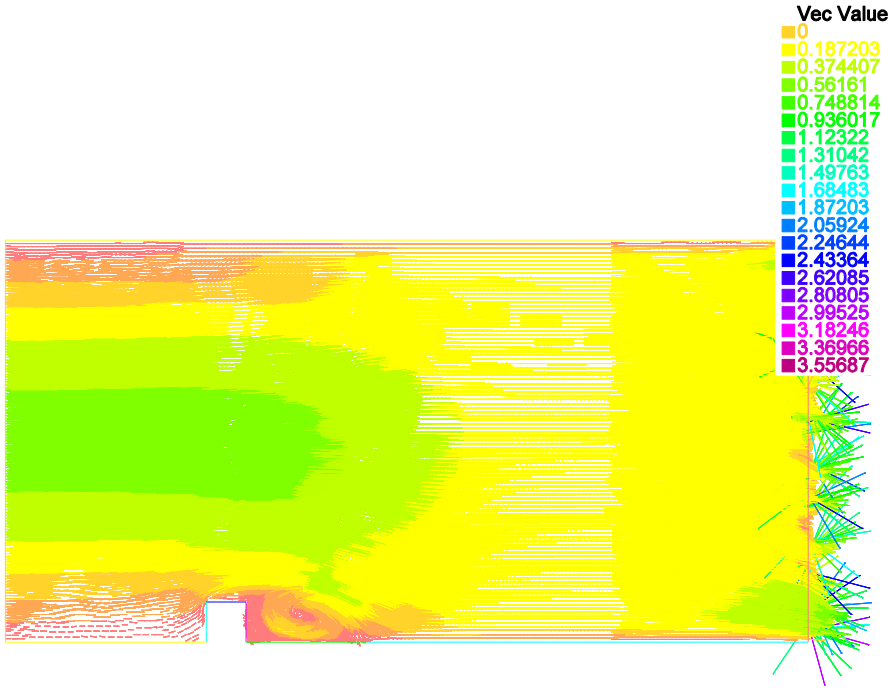

Figure 20. Step Problem, T = 10, Computed Velocity

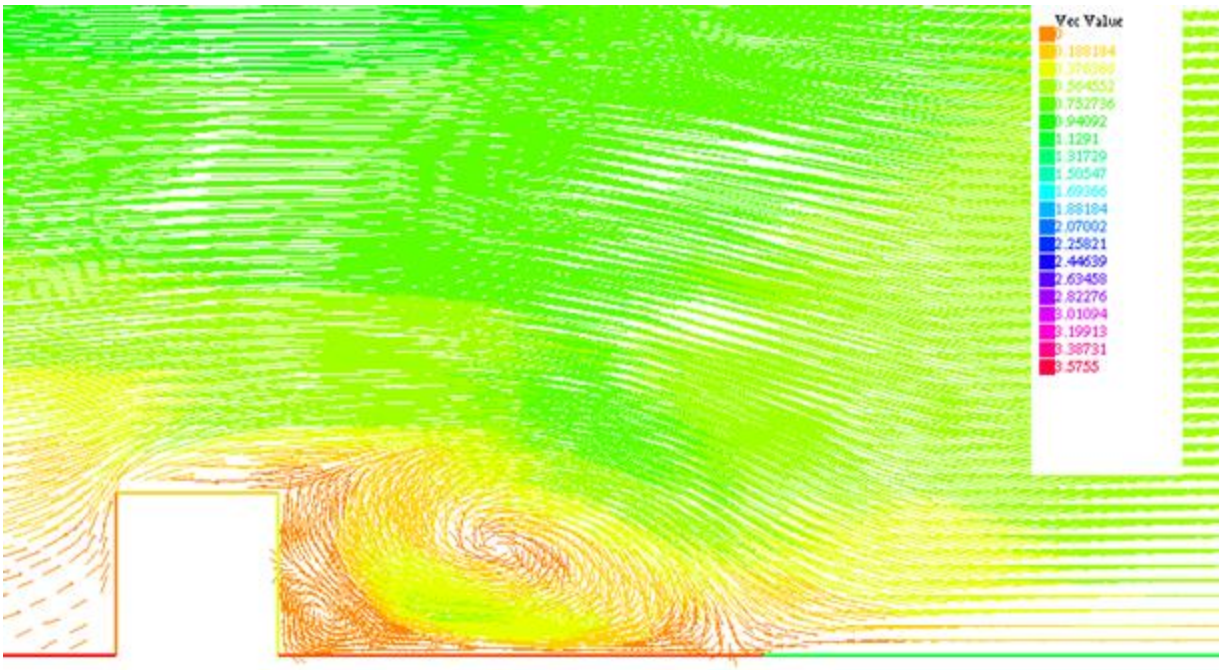

Figure 21. Step Problem, T = 10, Zoom in of Velocity near Step

Figure 22. Step Problem, T = 20, Computed Pressure

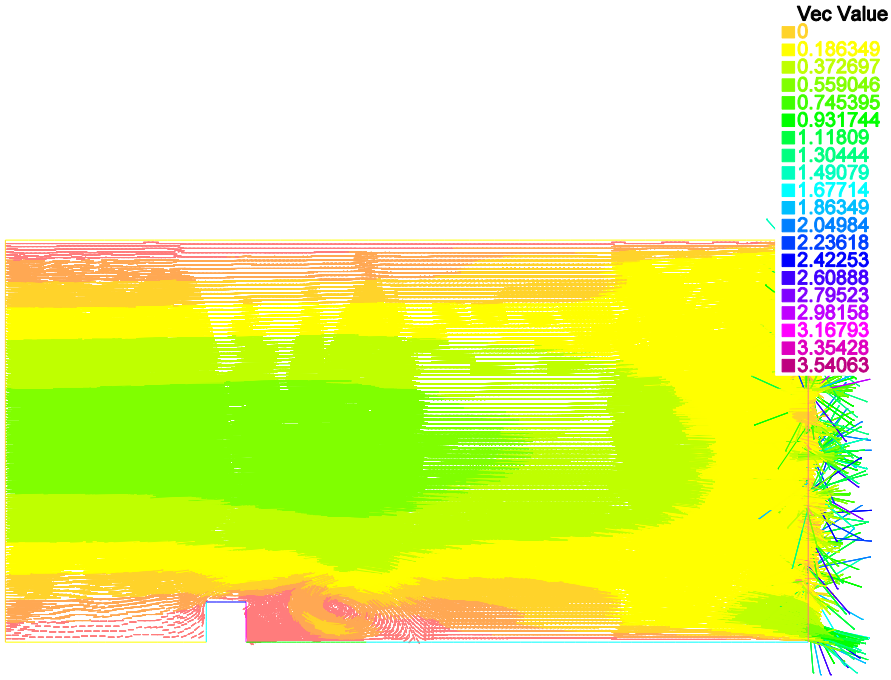

Figure 23. Step Problem, T = 20, Computed Velocity

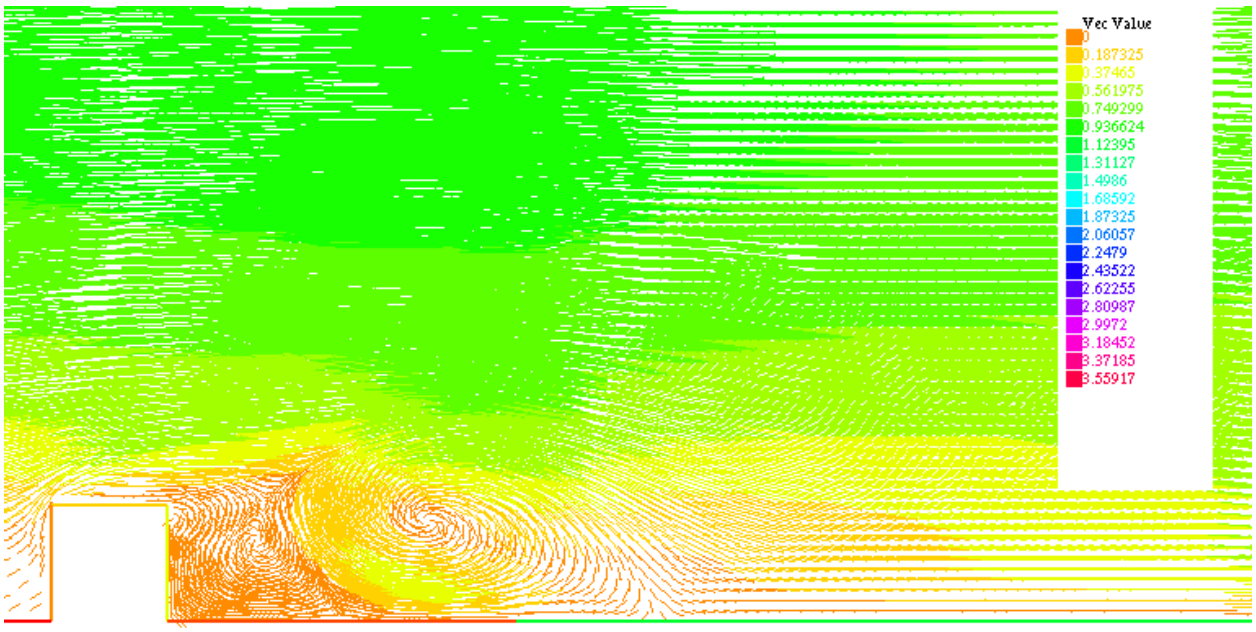

Figure 24. Step Problem, T = 20, Zoom in of Velocity near Step

Figure 25. Step Problem, T = 40, Computed Pressure

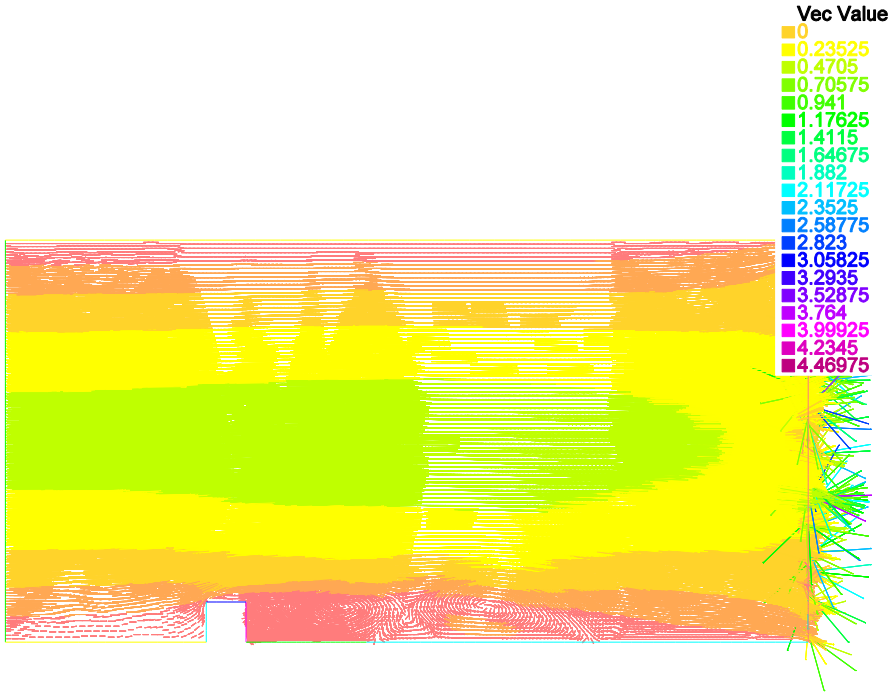

Figure 26. Step Problem, T = 40, Computed Velocity

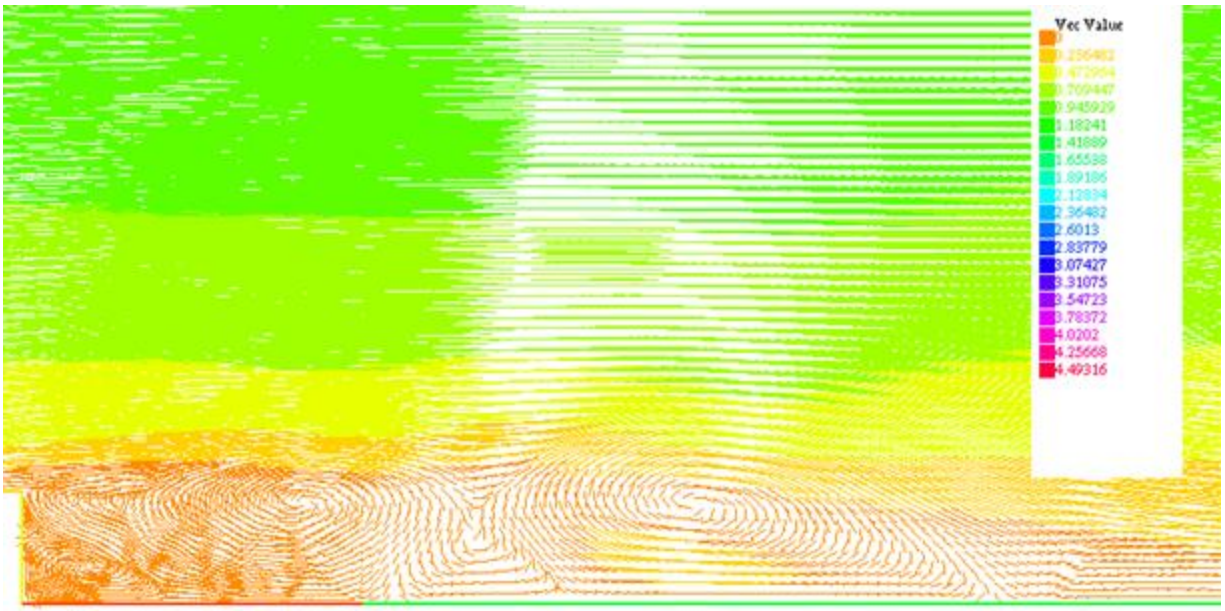

Figure 27. Step Problem, T = 40, Zoom in of Velocity near Step

When T = 10, a vortex forms on the right side of the step as expected. As time passes, more vortices form and travel farther towards x = 20. The vortices separate and trail down the x-axis. Note the odd behavior of the pressure and velocity at the outflow. This behavior is much worse than for the usual “do nothing” outflow condition for the NSE. This shows that adapting “do nothing” outflow condition to the model is a significant open problem.

6.0 SUMMARY AND CONCLUSIONS

A modification of the NSE for Newtonian fluids was made through introducing a kinetic energy modification and an eddy viscosity acting on the marginally resolved scales. This regularization is related to a model for non-Newtonian fluids. By using K41 theory and similar procedures as those for the Newtonian NSE, equations for the energy cascade and micro-scale were obtained in order to understand the phenomenology of the model. After this, it was possible to analyze how to choose the value of the model parameters in order to allow the micro-scale to be of the same order as the chosen mesh width.

A formula is derived which enforces the model micro-scale to equal the averaging radius ($\eta = \delta$). In computing practice, we almost always take the averaging radius equal to the mesh width. The question becomes now: Does forcing $\eta = \delta = \Delta x$ over diffuse large or transitional flow structures? We explore this in tests of the model applied to the forward-backward step and conclude that it is successful: separation of the trailing vortex does occur.

We also find the transition point between the correct $k^{-\frac{5}{3}}$ energy spectrum and the accelerated $k^{-\frac{11}{3}}$ energy decay. When $\eta = \delta$, so $\alpha \sim \delta^{\frac{4}{3}}$, we find the transition length scale is at approximately $O(\delta^{\frac{2}{3}})$, which is larger than $O(\delta)$. Thus, one improvement would be to use τ in the model to put this transition point at $\eta = \delta = \Delta x$.

With FreeFEM++, we were able to create a program to solve the non-Newtonian flow model with the Finite Element Method with a Backward Euler discretization. First, we observed the behavior of an exact known solution to the NSE. We saw the first order convergence that is expected. Lastly, we looked at the step problem and observed the behavior that we expected by examining the plots of the computed velocity near the step.

BIBLIOGRAPHY

[ALP04] M. ANITESCU, W. LAYTON AND F. PAHLEVANI, *Implicit for local effects and explicit for nonlocal effects is unconditionally stable,* ETNA 18 (2004), 174-187.

[KT07] V. K. KALANTAROV AND E. S. TITI, *Global attractors and determining modes for the 3D Navier-Stokes-Voight equations*, preprint 2007.

[K41] A. V. KOLMOGOROV, *The local structure of turbulence in incompressible viscous fluids for very large Reynolds number*, Dokl. Akad. Nauk SSR, 30:9-13, 1941.

[L09] W. LAYTON, *Introduction to the K41 Theory of Homogeneous, Isotropic Turbulence*, lecture notes, 2009.

[Lay09] W. LAYTON, *Some Basics of Vectors and Tensors in Fluid Mechanics: Vector and Tensor Analysis*, lecture notes, 2009.

[LLMNR07] A. LABOVSCHII, W. LAYTON, C. C. MANICA, M. NEDA, L. G. REBHOLZ, *The stabilized, extrapolated trapezoidal finite element method for the Navier-Stokes Equations*, C.M.A.M.E 198 (2009), 958-974, 2007.

[G89] M. D. GUNZBURGER, *Finite element methods for viscous incompressible flows: A guide to theory, practice, and algorithms*, San Diego, CA, Academic Press, Inc., 1989.

[O80] A. P. OSKOLKOV, *On the theory of Voight fluids*, Zap. Nauchn. Sem. LOMI, 96, (1980), 233-236.

[P83] O. PIRONNEAU, *The Finite Element Methods for Fluids*, John Wiley, Chichester, 1983.

[P00] S. POPE, *Turbulent Flows*, Cambridge University Press, 2000.

[P06] F. PAHLEVANI, *Sensitivity computations of eddy viscosity models with an application in drag computation*, International Journal for Numerical Methods in Fluids, 52, 381-392 (2006).

[Q93] L. QUARTAPELLE, *Numerical solution of the incompressible Navier-Stokes equations*, Birkhauser Verlag, 1993.

[W92] O. WALSH, *Eddy Solutions of the Navier-Stokes Equations*, 1992.