Analysis of Complex Viscoelastic Flows Using a Finite Element Method

by

Scott David Phillips

B.S., Purdue University (1999)

Submitted to the Department of Chemical Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2006

Signature of Author .....
Department of Chemical Engineering
February 1st, 2006

Certified by .....
Robert A. Brown
Professor of Chemical Engineering
Thesis Supervisor

Certified by .....
Robert C. Armstrong
Professor of Chemical Engineering
Thesis Supervisor

Accepted by .....
William M. Deen
Chairperson, Department Committee on Graduate Students

Analysis of Complex Viscoelastic Flows Using a Finite Element Method

by

Scott David Phillips

Submitted to the Department of Chemical Engineering
on February 1st, 2006, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

The field of computational fluid mechanics of viscoelastic flows has been well explored in the three decades since its inception. Still, even with the vast amount of work detailed in the literature, much remains to be done towards the improvement of models of viscoelastic fluids and the improvement of the numerical methods used to solve the set of governing equations. The work contained in this document is concentrated in the latter of these areas.

The main goal of this body of work is to develop a robust, efficient simulation package to model three-dimensional viscoelastic flows. In order to accomplish this goal, improvements to the numerical methods and equation formulation were necessary to help reduce the overall size of the equation set used to describe viscoelastic flows in three-dimensional geometries. In order to test their viability for use in reducing the overall size of the problem, concepts involving changing the formulation of the equations and the numerical methods used to find the solution to the equations were first implemented and analyzed in a previously developed two-dimensional finite element simulation package.

Implementation and analysis is discussed of a formulation change involving decoupling the calculation of the velocity gradient interpolant equation and the momentum and mass continuity equations in the DEVSS-G formulation. Two different decoupled methods for computing the velocity gradient, one using a global least squares approximation and the other a local patch algorithm, are explored. While both methods reduce to the true velocity gradient with mesh refinement, the patch algorithm is shown to require significantly more mesh refinement than the global least squares approximation to order to attain equivalent refinement of the solution. Comparison of the two methods taking into account the additional refinement requirements of the local patch algorithm makes clear the superiority of the decoupled global least squares approximation for calculation of the velocity gradient interpolant.

The versatility and robustness of the decoupled form of the DEVSS-G equations are

demonstrated through the addition and modification of the evolution equations describing the stress of the polymer as well as new physical quantities of the flow. A time-dependent, free-surface finite element method is developed in which an evolution equation derived from the kinematic boundary condition is used to describe the height of the free surface as a function of time. This new evolution equation is incorporated into the decoupled formulation by simply adding an additional step to the time integration to evaluate the change in the height of the surface during the current timestep and then updating the element locations in the deformable region of the mesh. Application of the new equation in this manner requires no knowledge of the direct dependence of the system on changes in the new quantity, allowing for quick and easy implementation.

Incorporation of more advanced constitutive equations is used as further example of the utility of the decoupled form of the DEVSS-G equations. For most continuum based constitutive equations, the dependence of the equations on the flow variables can be expressed explicitly, allowing for the coupled set of equations to be solved with Newton's method. However, the dependence of the stress on the flow cannot be explicitly written for more advanced constitutive equations such as those derived from kinetic theory or those employing Brownian dynamics, greatly hindering the performance of Newton's method in locating the solution to the system. As an illustrative example, incorporation into the decoupled equation formulation of the closed form of the Adaptive-Length-Scale model (ALS-C) is presented. Simulations are presented capturing for the first time the pressure drop enhancement with increasing viscoelasticity of the model of the flow of a Boger fluid in the 4:1:4 axisymmetric contraction-expansion geometry observed experimentally (Rothstein et al., 2001). Simulations of the flow of a 4-mode FENE-P model fluid within the geometry are also presented. Though its dependence on the flow field can be expressed analytically, the cost of computation using multimode models is typically prohibitive when using fully coupled equation sets as the overall problem size grows considerably with the addition of each new mode. Incorporation of the 4-mode model within the decoupled equation formulation adds relatively little computational cost to the overall calculation.

Employing the formulation and numerical methods developed herein, a new three-dimensional finite element package is described for simulating confined viscoelastic flows. To make the package more robust, a number of different boundary conditions are included for modeling different geometries used in polymer processing. To help reduce the burden associated with mesh refinement in three-dimensional meshes, a commercial meshing package utilizing o-grid refinement for localization of refinement is employed. Furthermore, to allow for computation of the large equation sets typically associated with three-dimensional geometries, a parallel implementation of the three-dimensional simulation package is developed based on the two-dimensional parallel method developed by Caola et al. ((Caola et al., 2001), (Caola et al., 2002)). Simulation results demonstrating the accuracy and performance of the method are presented.

As a test of the robustness of the three-dimensional method, simulations of the flow of Newtonian and Oldroyd-B fluids through a periodic, linear array of cylinders are

presented. Comparisons with previous calculations for the Oldroyd-B flow in an infinitely wide domain with no variations in the direction of the width show the same trend in the drag force on the cylinder with increasing viscoelasticity as well as in the size and shape of the vortices formed in the gap between the cylinders. The study of this flow includes effects of modeling the cross section of the flow as an infinite domain with no variation in the direction of the width, an infinite domain of periodic computational width, an infinite domain of periodic computational width and a symmetric flow above and below the cylinders, and a bounded domain with solid walls located 4 cylinder radii apart.

Thesis Supervisor: Robert A. Brown
Title: Professor of Chemical Engineering

Thesis Supervisor: Robert C. Armstrong
Title: Professor of Chemical Engineering

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

The field of study of flow of viscoelastic fluids has offered decades of rich and rewarding research opportunities, from work aimed at understanding the fundamental behavior of the systems driven by the academic community, to work aimed at design and optimization of production and processing technology driven by industry. While early research in the field focused on the most well known viscoelastic fluids of polymer solutions and polymer melts, more recent research has turned to systems of biological fluids such as solutions of DNA molecules which exhibit viscoelastic behavior in flow.

Simulation and modeling work aimed at better understanding and describing the flow of viscoelastic fluids has been an established field of research since its inception in the mid 1970's, evidenced by the numerous review articles written in the field [13], [24], [47], [48]. Since that time many strides has been made in modeling and simulation of these flows, but there is still much work to be done before the viscoelastic problem will be considered solved. The overarching goal of this body of work is to further the understanding of modeling and simulation of viscoelastic fluid flows.

As with the solutions of large problems in many fields of study, assumptions are often made to simplify the model of the physical process, making its solution tractable

by reducing the amount of time and effort needed to reach the solution [4], [29], [30]. The field of computational fluid mechanics of viscoelastic fluid flows is by no means an exception to this rule. The early simulation work in the field started with one- and two-dimensional simulations using coarse discretizations of relatively simple geometries. As time passed, improvements in computational power came about as well as improvements in the numerical algorithms employed to solve the problems. More and more detail could be added to simulate geometries of increasing complexity as well as complexity of the representation of dynamic behavior of the fluid and the physical description of the system.

Recently there has been significant effort toward extending the solution techniques used for two-dimensional domains to three-dimensional domains. While some of the motivation for this move was borne from the ability to compute problems of such size, much of the motivation has come from the loss of physics of assuming that a three-dimensional physical problem can be represented by a simplified one- or two-dimensional model. To help justify this statement, imagine a simple physical setting such as tangential flow in the annular region between two concentric cylinders, the so called Couette flow, Fig. 1-1. When the viscous forces of the fluid flow far outway the inertial forces and when considering only the region of the flow far away from ends of the cylinders, the problem can be reduced to a simpler two-dimensional geometry that is a cross-section of the cylinders perpendicular to their axes. Looking closely at one small section of this geometry, the curved walls of the cylinders can be viewed as two parallel plates, one stationary and the other moving at a linear velocity equal to the magnitude of the velocity of the moving cylinder. This approximation works well within the limit of very small differences in the inner and outer radii or the cylinders, but when this gap width increases, the approximation begins to break down. To the local fluid, the walls begin to appear curved, changing the behavior of the fluid as it flows through the annular region. Up to this point we have assumed that the direction parallel to the axes of the cylinders is a neutral direction in that no variations in behavior are noticeable in this direction. This assumption is applicable as long as we only observe the fluid at a point far away

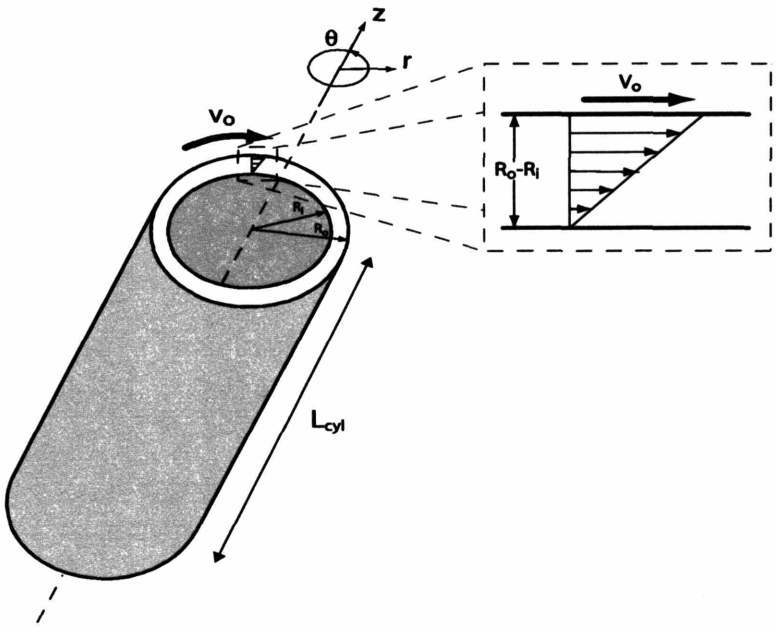


Figure 1-1: Example of the reduction in dimensionality of the modeling of the flow field in simple Couette flow from three dimensions shown in the main figure to one dimension shown in the inset. This reduction in dimesionality is appropriate only when $v_\theta = v_\theta(r)$.

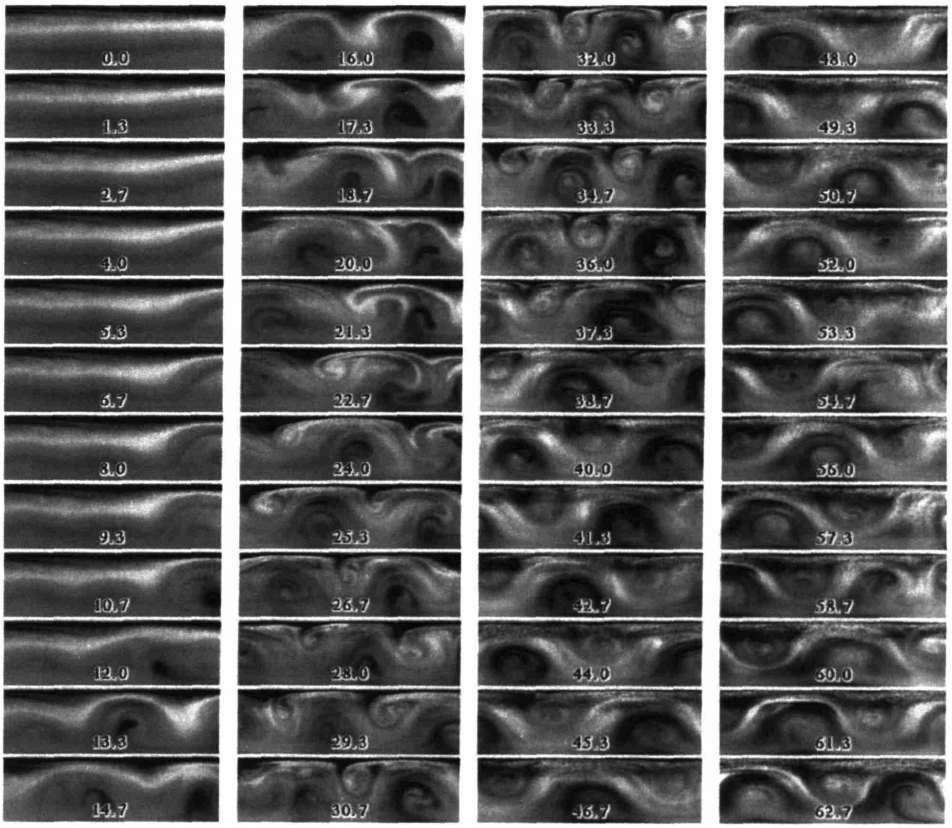


Figure 1-2: View of the gap cross-section of the low viscosity Boger fluid at De = 1.88, Re = 33.4 (co-rotating cylinders). Downwardly rolling vortices begin to form at 10 minutes. The direction of rotation of vortices along the inner cylinder is reveled by the clockwise motion of the comma shaped region of seed depletion in the vortex visible from 40 to 44 minutes. Reproduced from [9].

from either end of the cylinders as though the cylinders were infinitely long. However, as we approach either end of the cylinders we will see effects from the shear on the top and bottom bounding surfaces. The existence of a bounding wall appears to cause effects to propagate throughout the fluid and cause flow cells to develop that are at first three-dimensional steady and then become time-dependent as the viscoelasticity of the fluid is increased. The cells have been shown to travel up and down the axis of the cylinders, [9]. A time history of the formation of rolling cells of a Boger fluid within co-rotating cylinders is shown in Fig. 1-2.

Development of three dimensional structures from the growth of instabilities in the flow of viscoelastic fluids is in fact common throughout the literature. As shown by Evans and Walters ([31] and [32]) entry flows into a planar contraction of width to height ratio of 1 to 2 yields fully three-dimensional structures, another phenomena that could not possibly be captured with a two-dimensional simulation. Genieser [33] also explored the effects of contraction ratio on instabilities in the planar contraction. While the mechanism of the instability is not fully understood, it is apparent that the boundedness in the "neutral" direction is not likely the cause given that the instability exists for a wide range of width to height ratios. Another such instability that has been well studied is the flow around period array of cylinders of various inter-cylinder spacings [3], [71], [54], [57]. For the infinitely spaced case, that of an isolated cylinder in a channel, McKinley et al. [57] observed through experimentation three dimensional spatially periodic structures on the downstream side of the cylinder as illustrated in Fig. 1-3. Instabilities can also arise in unconfined flows such as those present in fiber spinning illustrated in Fig. 1-4. Here the flow is shown to progress from a stable column of fluid in (a), to a shark-skin instability starting in (b), and finally to gross melt fracture in (f) and (g).

Aside from the need for three-dimensional simulations to capture flow structures that arise from instabilities, there are also many industrially important problems in which the three-dimensional structure of the problem arises from the problem geometry, such as extrusion dies used in the fiber spinning industry and film casting industry. Traditional fiber spinning consists of extrusion and draw down of a cylindrical filament from a die. By assuming the filament and die are both axisymmetric, the model is reduced from three dimensions to two dimensions [71]. However, this is rarely the case in the fiber spinning industry. In some cases fibers of varying cross section are used to enhance properties such as touch and feel and even to enhance mass and heat transfer properties fabrics constructed of the fiber. Of all of the cross-sections pictured in Figs. 1-5 and 1-6, only the circular cross-section can be modeled by a two-dimensional axisymmetric simulation. The other fibers require three-dimensional simulations to capture the intricacy of the

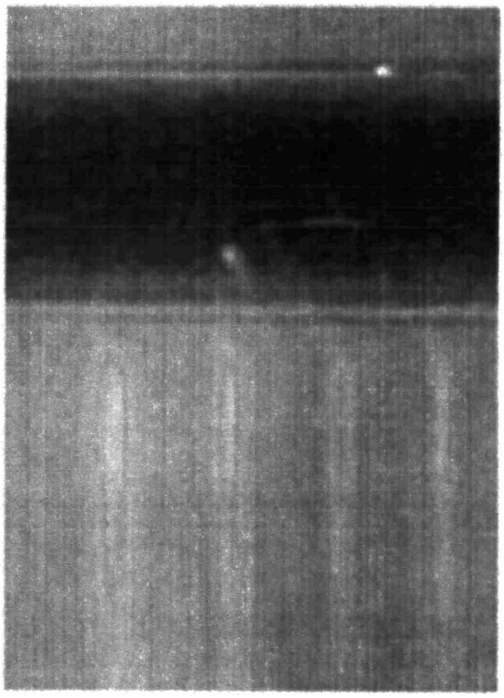


Figure 1-3: Steady, spatially periodic structure of a Boger fluid flowing past an isolated cylinder (black). Flow is from top to bottom. Reproduced from [57]

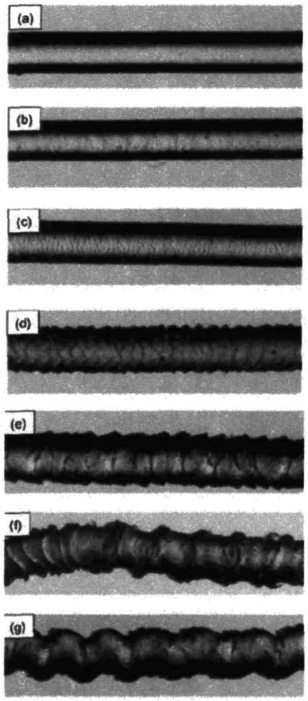


Figure 1-4: Example of an instability arising in the extrusion of a single strand polymer filament. Steady flow is shown in (a), evolving into a shark-skin instability in (b)-(e), and finally gross melt fracture in (f) and (g). Extrudates of a polymer melt at 70° C. Shear rates are (a) 1.36 s^{-1}, (b) 2.72 s^{-1}, (c) 6.81 s^{-1}, (d) 13.6 s^{-1}. (e) 34.1 s^{-1}, (f) 68.1 s^{-1}, and (g) 136 s^{-1}. Fiber is roughly 1 mm in diameter. Reproduced from [88].

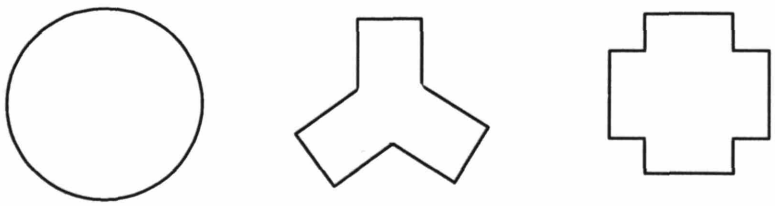


Figure 1-5: Various cross-sections of spun polymer fibers.

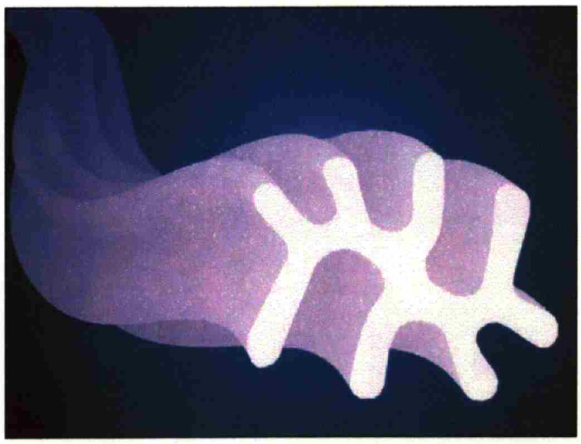


Figure 1-6: Artist rendition of a 4 DG Deep-Groove fiber. The cross-section is designed to have enhanced capillary action allowing for improved wicking properties as well as greatly increased surface to volume ratio over traditional fibers. Reproduced from [2]

fiber cross-section.

Even in the case of circular cross section fibers, because of the high processing rates used in industry, it is common practice to draw multiple fibers from a single die assembly and then draw the fibers together into a bundle. Die plates with three, five, or more die holes are used in these applications. While each die hole could behave as an independent contraction flow at relatively low processing rates with isolated lip vortices [8], it is unclear what would happen as the lip vortices from two adjacent die holes collide with

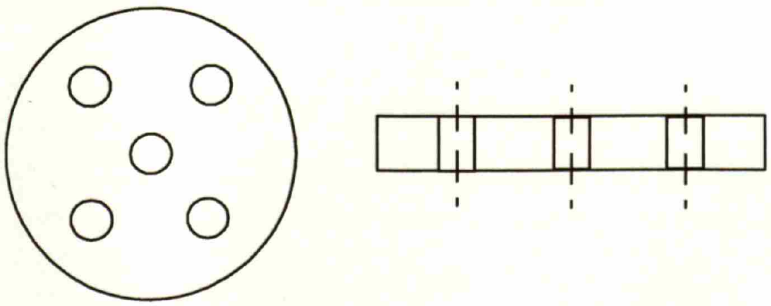


Figure 1-7: Possible die geometry and cross-section with five hole layout.

one another as the processing rates are increased. Certainly rich behavior beyond the modeling capacity of the two-dimensional case of the simple one-holed die is likely to occur.

It is clear that the need for three-dimensional modeling of the dynamics of viscoelastic fluids is important for the understanding and development of many applications. Work with the three-dimensional models to this point in the field has included mostly finite volume simulations on relatively coarse meshes as in [60], [87], and [89] with some attempts in finite elements on even coarser meshes [8]. More recently parallelization of the viscoelastic flow problem has been utilized to break up the calculation among a number of machines effectively reducing the overall time to reach solution or increasing the attainable calculation size. Application of a three-dimensional finite volume method can be found in [28] and that of a two-dimensional finite element method in [16].

1.2 Goals and Outline of Thesis

The overall goal of this body of work is to further develop the set of tools available for the analysis of viscoelastic flows. The main target for achievement of this goal is the development a three-dimensional modeling package for viscoelastic flows. Due to the large set of equations that describe typical three-dimensional viscoelastic flow systems, it is necessary to develop numerical methods and formulations that pursue the solution to the viscoelasitc flow system in the most optimal manner. To aide in this effort, the methods developed in this thesis take advantage of the techniques developed in a well-documented two-dimensional finite element package in terms of formulation of the equations and parallel solution of the resulting set of differential algebraic equations. To further reduce the overall size of the set of differential algebraic equations, modification of the formulation of the equations was studied and documented herein. Also, since a comprehensive package in terms of the physical description of the system is of interest, the use of additional evolution equations for added physical description and for new

and diverse descriptions of the polymer stress is addressed. For example, incorporation of evolution equations describing the motion of free surfaces within the computational domain and incorporation of a closed form of the Adaptive-Length-Scale model for improved description of the polymer stress are presented. Furthermore, since the size of the equation set resulting from sufficient refinement of the three-dimensional geometry with a finite element mesh can be quite sizable when compared to its two-dimensional counterpart, development of optimal parallel solution techniques to divide up the set of equations among multiple processors is presented. Finally, the power of the three-dimensional package is demonstrated on a complex viscoelastic flow known to exhibit a time-dependent elastic instability.

Since the field of modeling and simulation of viscoelastic flows is well established, a significant amount of background study is helpful before proceeding. The first couple of chapters of this thesis are dedicated to aid the reader in this endeavor. Chapter 2 discusses the physics of viscoelastic fluids in simple and complex flows. The chapter begins with the description of some of the more interesting flow phenomena that has been observed by experimentalists. The simple flows typically used to measure the performance of models for viscoelastic fluids by comparison of the computed and experimentally measured material functions are then described. Finally, the governing equations for viscoelastic fluid flow are given, starting with the conservation equations for the mass and momentum of the fluid and ending with continuum-based constitutive equations describing the relationship between the flow field and the stress of the fluid. While there have been a large number of constitutive equations developed over the history of modeling of viscoelastic fluids, equations described herein are only those most relevant to the work contained within this document and their most relevant predecessors.

Chapter 3 further aids the reader in the understanding of viscoelastic fluid flow modeling and simulation. This chapter is dedicated to the discussion of the numerical methods used to attack the daunting task of simulation of complex viscoelastic flows. The chapter begins with the description of the more recent history of the formulations of the viscoelas-

tic governing equations. The finite element method, the discretization workhorse in the field of viscoelastic flow simulation, is then discussed, with examples designed to help the reader understand how the governing equations are fit into this framework. The basis functions and elements are also included to help with understanding the implementation of the finite element method. Next, a discussion is included of the decoupled sub-problem formulation of the time-dependent set of equations with comparison to the direct computation of the steady-state set of equations. A brief treatment of time integration methods is included, with advantages and disadvantages of each given. The parallel solution method used within this work is then discussed in detail. Finally, to motivate the need for all of the advanced numerical methods included within this work, a sample problem size calculation is included for a model fiber spinning problem taken from the literature.

Chapter 4 is dedicated to the description of the decoupled velocity gradient interpolant formulation. This is a new method designed at reducing the overall size of the computational problem by taking advantage of the time-dependent form of the equations and the sub-problem formulation discussed in chapter 3. The chapter begins with discussion of the decoupled form of the equations and suggests two different forms of the equations used to compute the velocity gradient interpolant: a global least squares minimization and a local patch formulation. Measurement of the performance of these two formulations is presented in detail, the goal of which is to determine the method that provides the largest reduction in the overall problem size without loss of detail in the viscoelastic flow simulation. While the local patch formulation appears to offer great savings in overall problem size, as the Deborah number is increased, the velocity gradient field computed from the method deviates more and more from the accepted ideal of the fully coupled set of equations. To combat this increase in error, increased mesh refinement can be used, but at a significant cost to the overall calculation. The decoupled global least squares minimization formulation on the other hand shows virtually no deviation from the ideal solution and requires no further mesh refinement. The chapter

concludes with a sample problem size calculation comparing the fully coupled method with both the decoupled least squares and the local patch formulations, demonstrating the significant savings in problem size offered by the new decoupled global least squares formulation for the calculation of the velocity gradient interpolant.

Chapter 5 contains the development of the time-dependent free-surface viscoelastic finite element method for two-dimensional unconfined flows. The governing equations describing the flow of a viscoelastic fluid in an unconfined geometry are first presented. In addition to the governing equations given in the preceding chapters, the free-surface boundary conditions are now introduced along with the mapping equations used to update nodal positions in the deformable portion of the mesh. The numerical method used to attack this problem is then discussed, focussing on the implementation in the time-dependent, decoupled formulation. Though this development is carried out in the two-dimensional framework, it is intended to be applied to the full time-dependent, three-dimensional solver. The method outlined in this chapter is a demonstration of how an evolution equation describing some new physical aspect of the system is easily implemented in the decoupled, time-dependent formulation of the viscoelastic system of equations. Other equations that can be implemented in this manner included the energy evolution equation and the crystallization kinetics evolution equation. As a test of the implementation of the free-surface governing equations, simulations of the die-swell of a Giesekus fluid emanating from a contraction die are compared to a known two-dimensional steady-state method for a range of Deborah numbers, Weisenberg numbers, and Capillary numbers. Excellent agreement is demonstrated between the two methods for all cases.

Chapter 6 contains the simulations of the flow of a Boger fluid in the 4:1:4 contraction-expansion geometry. The models used to represent the Boger fluid are the closed version of the Adaptive Length Scale model and the 4-mode FENE-P model. Use of these models within the time-dependent decoupled framework helps to demonstrate the variety of constitutive equations that can be easily implemented in this framework. The rheology

of the fluid characterized in [64] and [65] is modeled, and the comparisons to the key rheological measurements are given in detail. The geometry used in the simulations is designed to model that used by Rothstein et al. [65]. The simulations of the flow in the 4:1:4 geometry with the ALS-C fluid model are the first to demonstrate pressure drop enhancement with increasing viscoelasticity, a well-known experimentally observed phenomenon. Qualitative growth of the vortex in the salient corner, parameterized into radial and axial location of the vortex center, as well as the reattachment length also agrees qualitatively with experimental findings. The simulation with the 4-mode FENE-P model also shows pressure drop enhancement and salient corner vortex growth, but the simulations prove to be difficult to converge as the viscoelasticity increases and do not show the dramatic trends that the ALS-C model simulations exhibit.

Chapter 7 details the three-dimensional finite element package for viscoelastic flows in confined geometries. This package is based on the two-dimensional method used in chapters 4-6. It builds upon the time-dependent, decoupled formulation and takes advantage of the decoupling of the velocity gradient interpolant equation as presented in chapter 4. To develop a method for robust use with many different physical geometries, implementation of a number of different boundary conditions was crucial. The boundary conditions implemented are given in detail. Next the elements and basis functions used in the three-dimensional package are given along with a description of the mesh generation software used. To test the accuracy of the method, simulations of Newtonian and Giesekus fluids flowing in pipe and duct geometries are compared to analytical solutions where available or otherwise to simulation generated from a well-tested two-dimensional method and to results published in the literature. Due to the large size of the finite element meshes needed to resolve flows in three-dimensional geometries, a parallel implementation of the three-dimensional package is detailed. Favorable performance of the package is demonstrated for the duct flow problem with increasing parallel machine size.

Chapter 8 contains simulations the flow of an Oldroyd-B fluid in a periodic, linear array of cylinders. The geometry is designed to match that used by Liu et al. [54] in

experimental identification a time-dependent instability of the flow of a Boger fluid and later used by Smith et al. [73] in a linear stability analysis of the flow. Comparisons are made of the flow fields and stress fields generated in an infinitely wide array described by a two-dimensional simulation and by a three-dimensional simulation with periodic boundaries on the side walls of the computational domain. The effects of adding solid side-walls to the geometry are also discussed. Finally, the flow structures arising in the infinite width array with a periodic computational domain of width 2, 3, and 4 units are explored and compared to the flow structure found by Smith et al..

Chapter 9 contains a summary of the work and results presented in this document. Finally, the author's views are included concerning the extensions of the work in this document and the possible exploitation of the techniques developed herein for further improving the simulation efforts for viscoelastic fluid flows.

Chapter 2

Physics of Viscoelastic Fluids

The physics associated with the flow of viscoelastic fluids is an area of research that is rich in interesting flow phenomena but is described by a set of governing equations that is difficult at best to solve. In this chapter, by way of justification of the above statement and as motivation into why the area of viscoelastic fluids is one of interesting research, some of the flow phenomena are first presented to give the reader a feel for the unique behavior of viscoelastic fluids in flow as compared to flows of the relatively simple Newtonian fluids. Simple flows that are commonly used to characterize the rheology of the viscoelastic fluids are then described. Finally, the equations governing the flow of viscoelastic fluids are presented.

2.1 Flow Phenomena

Many interesting and visually stimulating phenomena have been observed in the flow of fluids. A number of books are available with a broad array experimental visualizations from fluid flow. For phenomena focused mostly in the area of Newtonian flows, the reader is directed to the book assembled by Van Dyke [84]. An excellent collection of experimental visualizations for viscoelastic fluid flows is that compiled by Boger and Walters [12]. Here a few examples of the phenomena of viscoelastic flows that differ

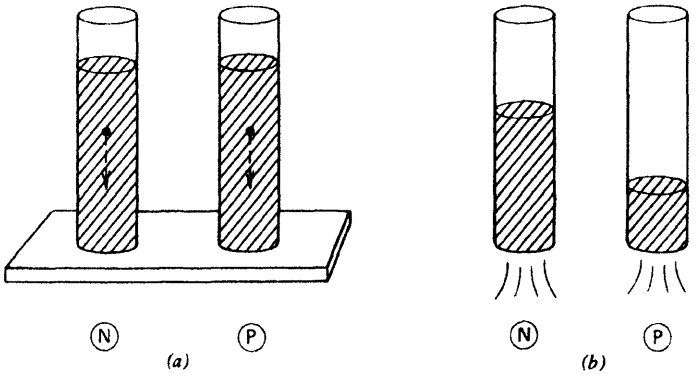


Figure 2-1: Tube flow and "shear thinning." In each part, the Newtonian behavior is shown on the left "N"; the behavior of a polymer on the right "P". (a) A tiny sphere falls at the save rate through each; (b) the polymer flows out faster than the Newtonian fluid. Reproduced from [10].

from Newtonian flows are presented.

2.1.1 Non-Newtonian Viscosity

One of the striking differences between Newtonian and Non-Newtonian fluids is the dependence that non-Newtonian viscosity can have on the shear-rate of the flow. The vast majority of non-Newtonian fluids that exhibit shear-rate dependent viscosity are what are known as shear-thinning fluids, or pseudoplastic fluids [10]. These fluids exhibit a decrease in viscosity with increasing shear-rate. Figure 2-1 portrays the flow of a Newtonian and a shear-thinning non-Newtonian fluid from a pipe. In fig. 2-1a, a tiny sphere is shown to fall in both fluids at the same rate with no discernible difference between the two fluids. In fig. 2-1b, the fluids are shown flowing from the pipes, with the shear-thinning fluid flowing faster due to the decrease in viscosity from the wall shearing the fluid.

A few fluids have been shown to exhibit behavior opposite to shear thinning, in which the viscosity increases with increasing shear rate [10]. These fluids are known as shear-

thickening fluids, or dilatant fluids.

A third class of non-Newtonian fluids are the fluids that exhibit a yield stress called viscoplastic fluids. Many examples of these fluids can be found, including common household items such as toothpaste or paint. The existence of the yield stress has been challenged on the ground that all fluids will flow given enough time has passed [10]. This being true, the yield stress is then a measure of the stress below which flow is negligible.

2.1.2 Normal Stress Effects

Another very significant difference between Newtonian and non-Newtonian fluids is the generation of differences in the normal stresses of non-Newtonian fluids in shear flow. In simple shear flow of polymer fluids, the first normal stress difference, the difference of the normal stress in the primary flow direction and the stress in the direction of variation in the flow field, is practically always negative [10]. Thisdifference can be thought of as an extra tension in the fluid along the streamlines of the flow in addition to the shear stress in the Newtonian case. This extra tension resists the flow, always wanting to return the material to its previous state. This simple explanation of tension along the streamlines has been used to explain a number of different experimentally witnessed phenomena [10].

The second normal stress difference, the difference in stress in the direction of variation of flow and the neutral direction in shear flow, is almost always positive in non-Newtonian fluids, though typically it is considerably smaller than the first normal stress difference [10]. This equates to a small extra tension exhibited in the neutral direction in shear flow.

One of the most graphic examples of the effects of normal stress differences is shown in the rod climbing experiment (fig. 2-2). In this experiment a vertical rod is placed in a beaker of Newtonian and non-Newtonian fluid. Upon rotation of the rod, the non-Newtonian fluid begins to "climb" up the rod against the forces of gravity, while the surface of the Newtonian fluid remains flat or is depressed if the fluid inertia is large enough. Simple explanation of the phenomena is given through the idea that the

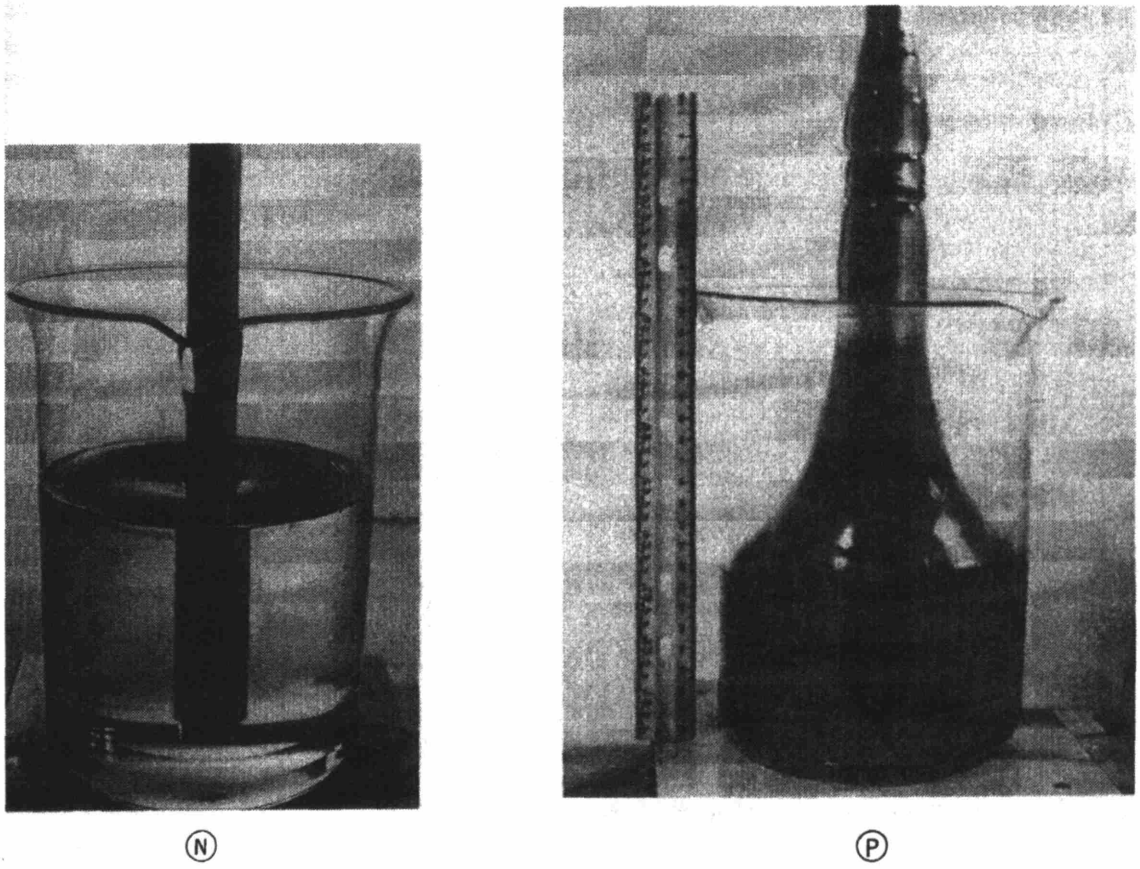


Figure 2-2: Fixed Cylinder with rotating rod. "N" The Newtonian liquid, glycerin, shows a vortex; "P" the polymer solution, polyacrylamide in glycerin, climbs the rod. The rod is rotated much faster in the glycerin than in the polyacrylamide solution. At comparable low rates of rotation of the shaft, the polymer will climb whereas the free surface of the Newtonian liquid will remain flat. Reproduced from [10].

streamlines surrounding the rod in closed loops all contain this extra tension. As the rod is turned, the tension along these concentric streamlines increases and strangulates the flow forcing it inward against the centrifugal force and upward against gravity. However, more in-depth analysis shows that both normal stress differences play a role in the fluid's behavior [10].

Another interesting phenomena attributed to the normal stress differences in non-Newtonian fluids is the die-swell effect (fig. 2-3). A Newtonian and a non-Newtonian fluid are shown here being extruded from a capillary die. The non-Newtonian fluid exhibits a significantly greater swell in the radius of the jet as compared to the Newtonian fluid. This extra swell can be attributed to the relaxation of the tension built up along the streamlines as the fluid emerges from the die [12].

2.1.3 Secondary Flows

Some flows of non-Newtonian fluids can exhibit additional weak secondary flow structures apart from the strong primary flow structure in flows where Newtonian fluids do not exhibit any secondary flow structure. The formation of the primary flow structure is roughly attributed to the viscous properties of the fluid, while the secondary flow structure is attributed to the inertial and elastic effects in the flow [10]. The elastic effects in the flow are typically opposite that of the inertial effects.

One interesting experiment in which a sphere is rotated in a sea of fluid shows significant differences in the between the Newtonian and non-Newtonian fluids. In the Newtonian case, the primary flow is shown by streamlines of concentric circles carrying fluid around the rotating sphere. A weak secondary flow driven by inertial effects is also seen, pulling fluid towards the sphere near the axis of rotation [10]. The analogous experiment was performed by Giesekus [35] for a 5% solution of polyacrylamide. He found the same primary flow, but with a secondary flow carrying fluid toward the sphere in the equatorial plane and away from the sphere near the axis of rotation.

Another interesting experiment in which differences in the secondary flow are seen is

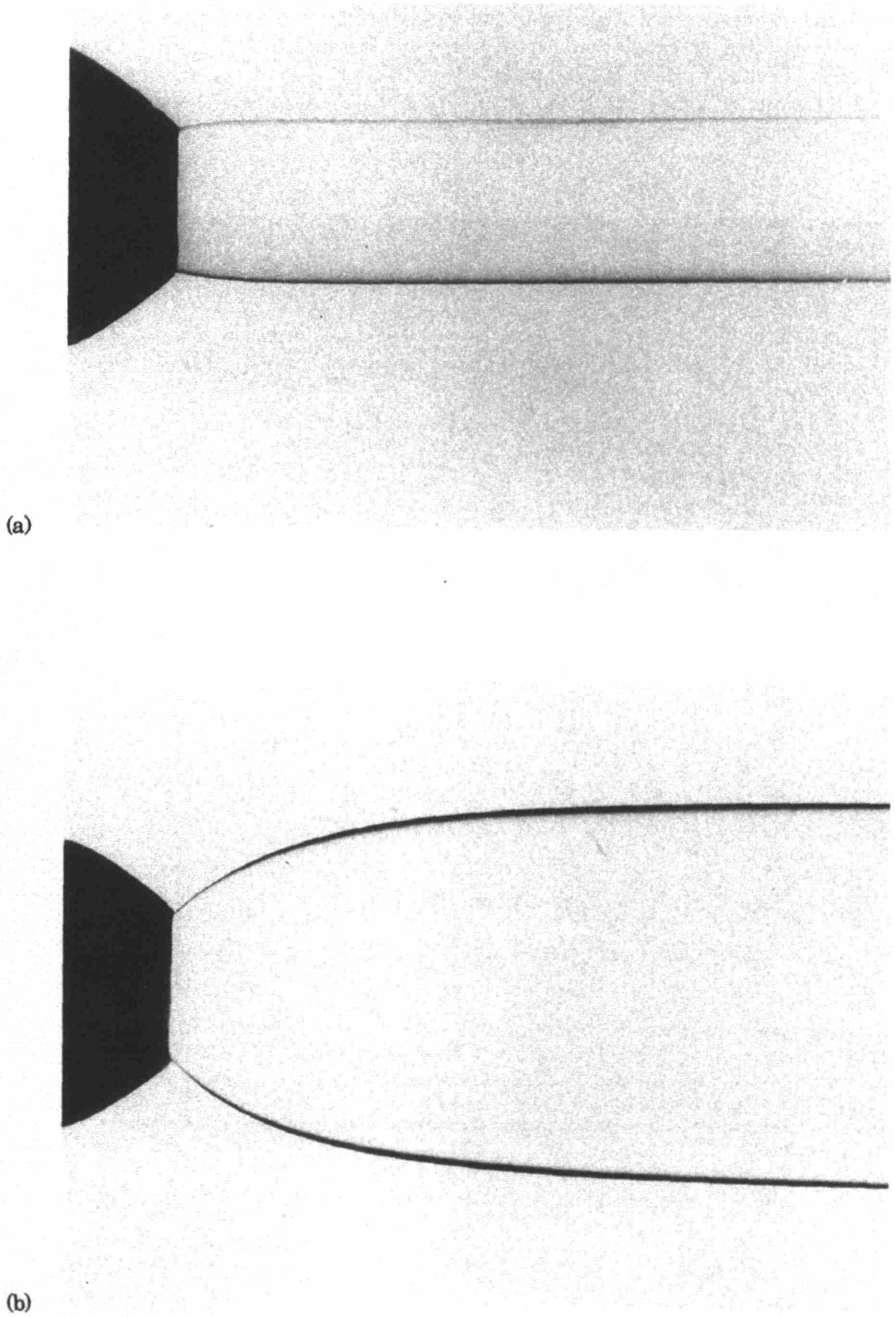


Figure 2-3: Die swell for liquid extruded into a neutrally-bouyant medium constructed from a low viscosity silicone oil and carbon tetrachloride solution of matching density to the extruded flow medium. (a) Newtonian liquid of viscosity 11.6 Pa s being extruded (Re=0.001). (b) Boger fluid of viscosity 11.4 Pa s being extruded (Re=0.0009, We=0.272). Reproduced from [12]

the cylindrical tank with a rotating lid experiment [10]. For the Newtonian case, the flow of the fluid is tangential to the rotation of the lid and decreases in strength moving away from the lid. A secondary flow is seen in which centrifugal force near the surface of the lid throws fluid radially outward, perpendicular to the primary flow. In the analogous experiment with a non-Newtonian fluid, a secondary flow in the exact opposite direction is seen as fluid is drawn inward at the surface of the lid where the normal stresses of the primary flow have acted to counter the effects of centrifugal force.

2.2 Simple Flows and Material Functions

Two types of simple flows are commonly used to measure experimentally the material functions of viscoelastic fluids. These two types are shear flows and shearfree flows. Measurements of material properties are most conveniently done in flows that are homogeneous, where the velocity gradient tensor does not vary with position. Both the steady and unsteady, startup and cessation, shear and shearfree flows are of importance in fluid rheological characterization. Detailed discussion of these flows and there use in fluid rheological characterization can be found in [10].

2.2.1 Shear Flow

The velocity profile for shear flow is illustrated for simple shear flow by

$$v_x = \dot{\gamma}_{yx} y \tag{2.1a}$$

$$v_y = 0 \tag{2.1b}$$

$$v_z = 0 \tag{2.1c}$$

where $\dot{\gamma}_{yx}$ is the shear rate. The flow is shown schematically in fig. 2-4. Particles in shear flow that are separated by a distance $y > 0$ will become separated from one another an amount linearly proportional to the amount of time over which the shear flow

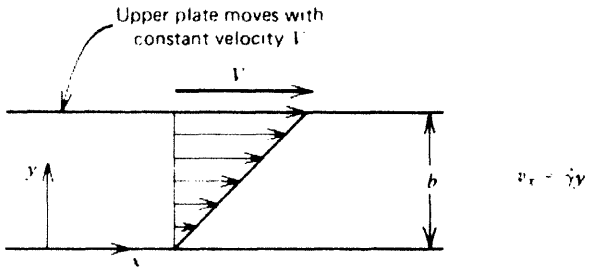


Figure 2-4: Steady simple shear flow with shear rate $\dot{\gamma} = V/b$. Reproduced from [10].

is applied, for large times. Physically, this flow can be thought of as a deck of cards where the top card is slid at a velocity V and each card below the top card moves with a linearly decreasing velocity with the bottom card remaining stationary [10]. The deformation of a material cube in steady shear flow shown in fig. 2-5 helps in visualization of this flow.

The material functions for steady shear flow are defined as follows. The viscosity for shear flow is defined as

$$\tau_{yx} = -\eta\left(\dot{\gamma}\right)\dot{\gamma}_{yx} \tag{2.2}$$

The normal stress coefficients for steady shear flow are defined as

$$\tau_{xx} - \tau_{yy} = -\Psi_1\left(\dot{\gamma}\right)\dot{\gamma}_{yx}^2 \tag{2.3}$$

$$\tau_{yy} - \tau_{zz} = -\Psi_2\left(\dot{\gamma}\right)\dot{\gamma}_{yx}^2 \tag{2.4}$$

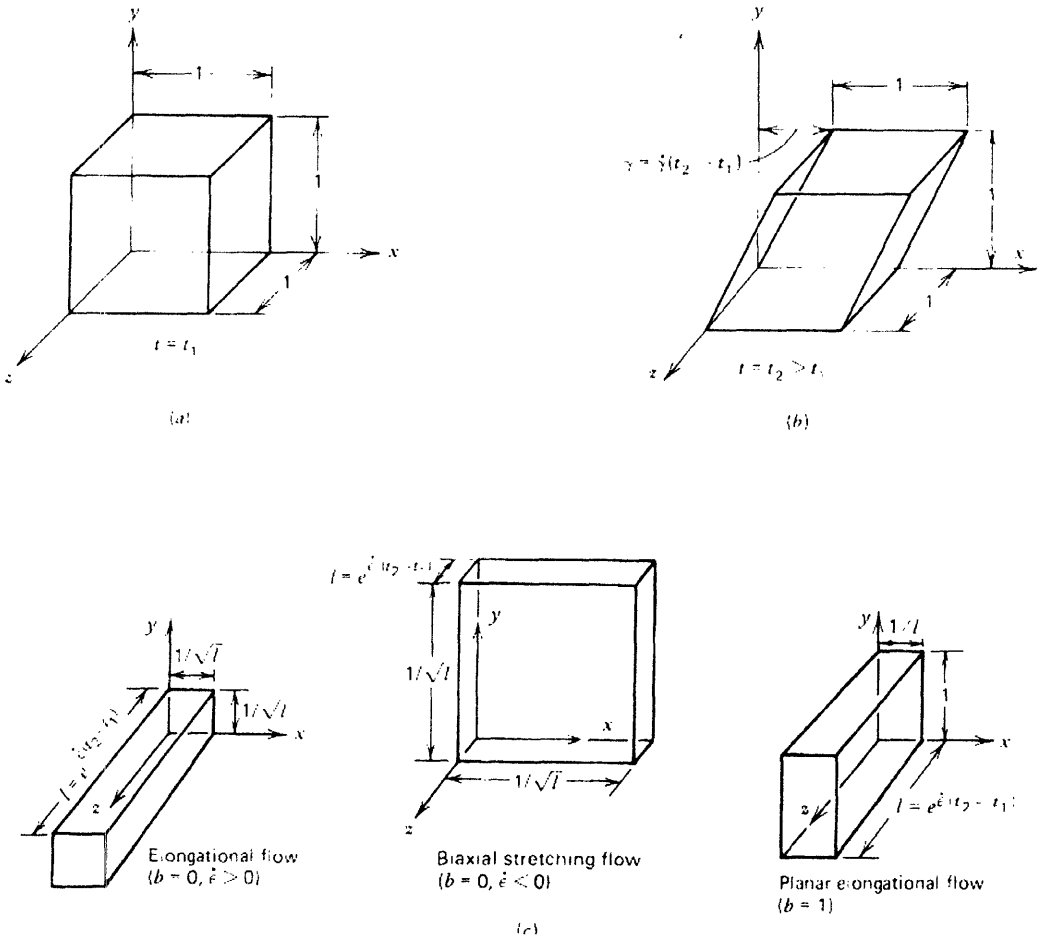


Figure 2-5: Deformation of (a) unit cube of material from time t_1 to t_2 ($t_2 > t_1$) in (b) steady simple shear flow and (c) three kinds of shearfree flow. The volume of material is preserved in all of these flows. Reproduced from [10]

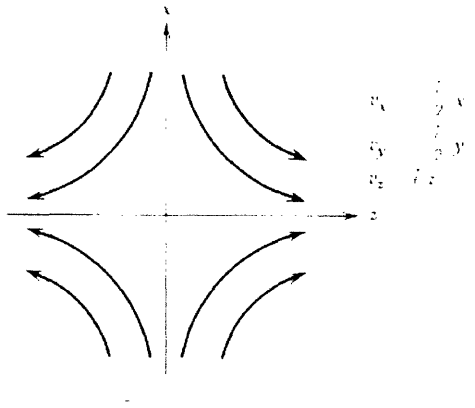


Figure 2-6: Steady elongational flow (shearfree flow with $b = 0$ and $\dot{\epsilon} > 0$). Reproduced from [10].

2.2.2 Shearfree Flow

The velocity profile for simple shearfree flow is given by

$$v_x = -\frac{1}{2}\dot{\epsilon}\,(1+b)\,x \tag{2.5a}$$

$$v_y = -\frac{1}{2}\dot{\epsilon}\,(1-b)\,y \tag{2.5b}$$

$$v_z = \dot{\epsilon} z \tag{2.5c}$$

where $\dot{\epsilon}$ is the elongational rate and $0 \leq b \leq 1$. Choices for the two parameters yield special cases of shearfree flow. Uniaxial elongational flow is given by the choice $b = 0$ and $\dot{\epsilon} > 0$. Biaxial stretching flow is given by the choice $b = 0$ and $\dot{\epsilon} < 0$. Planar elongational flow is given by the choice $b = 1$. The flow for the $b = 0$, $\dot{\epsilon} > 0$ case is shown schematically in fig. 2-6. Particles in the shearfree flow move relative to one another at an exponential rate, a rate much faster than the linear rate of separation in shear flow. As a loose comparison, biaxial stretching flow can be pictured as starting with a spherical ball of dough and then pressing the ball of dough into a round flat disk, like a pizza crust. While this is not a true elongational flow, this will hopefully help

the reader to visualize the flow. Deformations of a cube of material in the three special cases of elongational flow are shown in fig. 2-5.

The material functions for shearfree flow are defined as follows. For steady shearfree flow two viscosity functions $\bar{\eta}_1$ and $\bar{\eta}_2$ are defined as

$$\tau_{zz} - \tau_{xx} = -\bar{\eta}_1(\dot{\epsilon}, b)\,\dot{\epsilon} \tag{2.6}$$

$$\tau_{yy} - \tau_{xx} = -\bar{\eta}_2(\dot{\epsilon}, b)\,\dot{\epsilon} \tag{2.7}$$

For uniaxial extension and biaxial stretching where $b = 0$, $\bar{\eta}_2 = 0$ and $\bar{\eta}_1$ becomes equal to the elongational viscosity, $\bar{\eta}(\dot{\epsilon})$.

2.3 Governing Equations

A set of partial differential equations is used to describe the flow of a viscoelastic fluid. This set consists of equations describing the conservation of momentum and mass, as well as a constitutive equation describing the relationship between the flow kinematics and the polymer stress. Detailed derivation and discussion of the conservation equations can be found in [11] and [27] and discussion of the constitutive equations can be found in [10].

2.3.1 Conservation Equations

The equations of mass continuity and momentum conservation govern the flow of all fluids. The equation describing the conservation of mass, the continuity equation, is given by

$$\frac{\partial \rho}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \mathbf{v}) = 0 \tag{2.8}$$

where ρ is the density of the fluid, t is the time, $\boldsymbol{\nabla}$ is the gradient operator, and $\mathbf{v}$ is the velocity vector of the fluid. Using the incompressibility assumption and non-dimensionalizing with the characteristic length and velocity scales, the continuity equa-

tion becomes

$$\bar{\boldsymbol{\nabla}} \cdot \bar{\mathbf{v}} = 0 \tag{2.9}$$

The quantities with overbars are the non-dimensional forms.

The equation describing conservation of momentum is given by

$$\frac{\partial}{\partial t}(\rho \mathbf{v}) + \boldsymbol{\nabla} \cdot (\rho \mathbf{v}\mathbf{v}) = \rho \mathbf{g} - \boldsymbol{\nabla} \cdot \boldsymbol{\pi} \tag{2.10}$$

where $\mathbf{g}$ is the gravity vector and $\boldsymbol{\pi}$ is the total stress tensor. Assuming incompressibility and that the effects of gravity are negligible in the modeled geometries, and rewriting the total stress of the fluid as the sum of the pressure and the deviatoric stress, $\boldsymbol{\pi} = \mathbf{I}P + \boldsymbol{\tau}$, where $\mathbf{I}$ is the identity tensor, the momentum balance takes the form

$$\rho \frac{D\mathbf{v}}{Dt} = -\boldsymbol{\nabla} P - \boldsymbol{\nabla} \cdot \boldsymbol{\tau} \tag{2.11}$$

where $\frac{D}{Dt} \equiv \frac{\partial}{\partial t} + \mathbf{v} \cdot \boldsymbol{\nabla}$ is the substantial derivative, P is the pressure of the fluid, and $\boldsymbol{\tau}$ is the deviatoric stress. The characteristic length and velocity scales are represented as h^* and v^*. The characteristic viscous stress, $P^* = \mu v^*/h^*$, is chosen to scale the pressure and stress terms and the characteristic time scale is chosen as the process timescale, $t^* = h^*/v^*$, resulting in the non-dimensional form

$$\mathrm{Re} \frac{D\bar{\mathbf{v}}}{D\bar{t}} = -\bar{\boldsymbol{\nabla}} \bar{P} - \bar{\boldsymbol{\nabla}} \cdot \bar{\boldsymbol{\tau}} \tag{2.12}$$

where $\mathrm{Re} \equiv \frac{v^* h^* \rho}{\mu}$ is the Reynolds number. Often in polymer flows, $\mathrm{Re} << 1$. In this case, the momentum balance reduces to the Stokes form,

$$\mathbf{0} = -\bar{\boldsymbol{\nabla}} \bar{P} - \bar{\boldsymbol{\nabla}} \cdot \bar{\boldsymbol{\tau}} \tag{2.13}$$

2.3.2 Constitutive Equations

The constitutive equation is a relationship between the deformation of the material or fluid and the stresses generated within it. The simplest constitutive equation is that of a Newtonian fluid, namely

$$\boldsymbol{\tau} = -\eta\dot{\boldsymbol{\gamma}} \tag{2.14}$$

where η is the constant fluid viscosity and $\dot{\boldsymbol{\gamma}} \equiv \boldsymbol{\nabla}\mathbf{v} + \boldsymbol{\nabla}\mathbf{v}^T$ is the rate-of-strain tensor. Since the fluid viscosity is constant for a given pressure and temperature in a Newtonian fluid, the constitutive equation is a simple linear relationship between stress and rate-of-strain in the fluid.

The simplest of the non-Newtonian fluid models is the Generalized Newtonian Fluid model, represented by the constitutive equation

$$\boldsymbol{\tau} = -\eta\left(\dot{\gamma}\right)\dot{\boldsymbol{\gamma}} \tag{2.15}$$

where η is now a function of the shear rate, defined as $\dot{\gamma} = \frac{1}{2}\sqrt{\Pi_{\dot{\gamma}}}$, where $\Pi_{\dot{\gamma}}$ is the second invariant of the rate-of-strain tensor. Various models for fitting experimental $\eta\left(\dot{\gamma}\right)$ data have been proposed. The five parameter Carreau-Yasuda model has proven useful for numerical simulations, representing the dependence of viscosity on the rate-of-strain tensor as

$$\frac{\eta - \eta_\infty}{\eta_0 - \eta_\infty} = \left[1 + \left(\lambda\dot{\gamma}\right)^a\right]^{(n-1)/a} \tag{2.16}$$

where η_0 is the zero-shear-rate viscosity, η_∞ is the infinite-shear-rate viscosity, λ is a time constant, n is the power-law exponent, and a is a dimensionless parameter that describes the transition region between the zero-shear-rate region and the power-law region. Another relationship for $\eta\left(\dot{\gamma}\right)$ more commonly used in industry is the Power-Law model given by

$$\eta = m\dot{\gamma}^{n-1} \tag{2.17}$$

where m and n are parameters used to fit experimental viscosity data. It is useful to note

that when $m = \eta$ and $n = 1$ the Newtonian constitutive equation is recovered. Also, if $n < 1$, the fluid exhibits shear-thinning behavior, and when $n > 1$ the fluid exhibits shear-thickening behavior. Other empirical forms of $\eta(\dot{\gamma})$ for the Generalized Newtonian Fluid model can be found in [10].

While the Generalized Newtonian Fluid model has proven useful in a number of applications, use of this model is not always appropriate. The model becomes invalid when elastic effects become important in the flow and is strictly only applicable to steady-state shear flows [10]. To determine when this occurs, the dimensionless quantity known as the Deborah number is defined as

$$\mathrm{De} = \frac{\lambda}{t_{flow}} \tag{2.18}$$

where λ is the characteristic time of the fluid, typically taken as the timescale associated with the longest relaxation time of the fluid, and t_{flow} is the timescale of the flow. When $\mathrm{De} << 1$ elastic effects can be neglected and the Generalized Newtonian Fluid model is applicable. When De is of $O(1)$ both viscous effects and elastic effects are important and the Generalized Newtonian Fluid model is no longer valid. Accurate representation of these viscoelastic fluids requires a model that takes into account both viscous and elastic effects within the fluid.

One of the earliest known models used in the study of viscoelastic fluids was that of Maxwell. In his model he described the stress of the fluid as a combination of effects arising from the both the viscous and the elastic responses of the fluid. The Maxwell model is given by

$$\boldsymbol{\tau} + \lambda_1 \frac{\partial}{\partial t}\boldsymbol{\tau} = -\eta_0 \dot{\boldsymbol{\gamma}} \tag{2.19}$$

where $\lambda_1 = \mu/G$ is the relaxation time of the fluid with μ being the viscosity and G being the elastic modulus of the fluid. The Maxwell model comes from a class of models known as linear viscoelastic models. Its range of validity is limited to flows with infinitesimally small displacement gradients [10].

To improve upon the linear viscoelastic models and include non-linear effects in the models, a Taylor expansion about the Newtonian fluid is used in the class of models known as the retarded motion expansions. The expansions make use of the nth convected derivative of the rate-of-strain tensor $\gamma_{(n+1)}$,

$$\gamma_{(1)} \equiv \dot{\gamma} \tag{2.20a}$$

$$\gamma_{(n+1)} = \frac{D}{Dt}\gamma_{(n)} - \left\{(\nabla \mathbf{v})^T \cdot \gamma_{(n)} + \gamma_{(n)} \cdot (\nabla \mathbf{v})\right\} \tag{2.20b}$$

The nth convected derivative is defined in such a way that it is independent of the superposed rigid rotations [10]. Stress in the retarded motion expansion is then expressed as

$$\tau = -\left[b_1\gamma_{(1)} + b_2\gamma_{(2)} + b_{11}\left\{\gamma_{(1)} \cdot \gamma_{(1)}\right\}\right] + \text{higher order terms} \tag{2.21}$$

where b_1, b_2, and b_{11} are material parameters often called the retarded-motion constants. The order of the expansion is given by the highest order of the velocity gradient terms. The expansion shown in eq. 2.21 is the second-order expansion and the fluid modeled by this equation is referred to as a second-order fluid. While the retarded-motion expansions have proved useful in yielding a great deal of insight as to the effects of elasticity of the fluid, they are not widely applicable to industrially relevant problems as they are limited to modeling flows that are slow or slowly varying [10].

Derivation of differential models that are valid for arbitrary, time-dependent flows requires the introduction of the upper convected derivative of the stress tensor $\tau_{(1)}$,

$$\tau_{(1)} = \frac{D}{Dt}\tau - \left\{(\nabla \mathbf{v})^T \cdot \tau + \tau \cdot (\nabla \mathbf{v})\right\} \tag{2.22}$$

Replacing the time derivative of the stress tensor in the Maxwell model with the upper convected derivative yields the upper convected Maxwell model,

$$\tau + \lambda_1\tau_{(1)} = -\eta_0\dot{\gamma} \tag{2.23}$$

a widely used model in computation of viscoelastic flows due to its simplicity [10].

The Oldroyd B fluid model is obtained by adding the first convected derivative of the rate-of-strain tensor to the model, a quasi-linear viscoelastic model known as .

$$\boldsymbol{\tau} + \lambda_1 \boldsymbol{\tau}_{(1)} = -\eta_0 \left(\boldsymbol{\gamma}_{(1)} + \lambda_2 \boldsymbol{\gamma}_{(2)} \right) \tag{2.24}$$

where λ_2 is the retardation time of the fluid. The stress in the model can be rewritten as the sum of stresses from the solvent and polymer components of the fluid yielding the expression

$$\boldsymbol{\tau} \equiv \boldsymbol{\tau}_s + \boldsymbol{\tau}_p \tag{2.25a}$$

$$\boldsymbol{\tau}_s \equiv -\eta_s \dot{\boldsymbol{\gamma}} \tag{2.25b}$$

$$\boldsymbol{\tau}_p + \lambda_1 \boldsymbol{\tau}_{p(1)} = -\eta_p \dot{\boldsymbol{\gamma}} \tag{2.25c}$$

where the zero-shear-rate viscosity has been rewritten as $\eta_0 = \eta_s + \eta_p$ and the retardation time has been rewritten as $\lambda_2 = \lambda_1 \frac{\eta_s}{\eta_0}$.

While the Oldroyd-B fluid model and other quasi-linear viscoelastic models are valid for arbitrary time-dependent flows, they still fail to predict real material properties such as decrease of viscosity and first normal stress coefficient with increasing shear rate in steady shear flow. For this, nonlinear differential constitutive equations are necessary. One such model that has considerable diversity in its ability to predict rheological phenomena is the Giesekus model [10]. Its form is similar to that of the Oldroyd-B model, with the addition of a quadratic term in stress. The Giesekus model is given as

$$\boldsymbol{\tau} = \boldsymbol{\tau}_s + \boldsymbol{\tau}_p \tag{2.26a}$$

$$\boldsymbol{\tau}_s = -\eta_s \dot{\boldsymbol{\gamma}} \tag{2.26b}$$

$$\boldsymbol{\tau}_p + \lambda_1 \boldsymbol{\tau}_{p(1)} - \alpha \frac{\lambda_1}{\eta_p} \{ \boldsymbol{\tau}_p \cdot \boldsymbol{\tau}_p \} = -\eta_p \dot{\boldsymbol{\gamma}} \tag{2.26c}$$

where α is the mobility factor. Note that at $\alpha = 0$, the Giesekus model reduces to the

Oldroyd-B fluid model.

In steady shear flow, unlike the Oldroyd-B fluid model, the Giesekus model is capable of capturing shear thinning behavior with $\eta(\dot{\gamma}) \approx \dot{\gamma}^{-1}$ and $\Psi_1(\dot{\gamma}) \approx \dot{\gamma}^{-3/2}$ at high shear rates. The model also predicts a constant second normal stress difference. For low shear rates, $\Psi_2/\Psi_1 \approx -\alpha/2$, which is physically acceptible for concentrated solutions and polymer melts if α is between 0.1 and 0.5. In steady elongational flow the model predicts an elongational viscosity of $\bar{\eta} = 2\eta_0/\alpha$ at high strain rates. In transient shear flows, the model overshoots both viscosity and first normal stress coefficients. Detailed information on the rheological behavior of the Giesekus model can be found in [10].

Chapter 3

Numerical Methods for Viscoelastic Flows

3.1 Formulations of the viscoelastic governing equations

Many different formulations have been used to solve the governing equations of viscoelastic flow systems in order to deal with numerical difficulties introduced by both the momentum equation and the constitutive equation. At first inspection, the numerical representations of the equations appear to fall into the two categories of an elliptic partial differential equation in the case of the momentum equation and a hyperbolic partial differential equation in the case of the types of constitutive equations used in this field. However, the straight forward numerical formulation of the equations turns out to change characteristics with variations in flow fields and system parameters. This change is known to be a numerical artifact, as it is possible to prove that the continuity/momentum equation pair must be elliptical. Therefore, much effort has been put forth to formulate the numerical implementation of the equations in such a way as to preserve the elliptic nature of the momentum equation without sacrificing numerical

stability or other such properties of the system.

One of the early formulations for the viscoelastic flow equations was the Explicitly Elastic Momentum Equation, EEME, formulation which ensured the elliptic nature of the numerical momentum equation by taking the divergence of the upper convected Maxwell constitutive equation together with a modified pressure and substituting both into the set of equations. The result is a term in the momentum equation that can be shown to maintain its ellipticity independent of the flow conditions.

The treatment of solutions and low molecular weight species in melts by the inclusion of a Newtonian term led to replacing the UCM constitutive equation with the Oldroyd-B constitutive equation, where the total stress is a summation of polymer and solvent stress. Substituting this total stress into the momentum equation results in a form in which β, the ratio of the solvent viscosity to the total viscosity, multiplies the elliptic term. In the limit of $\beta \rightarrow 0$, the ellipticity of the momentum equation is lost. Furthermore, due to the numerical approximation of the elliptic term, even with $\beta << 1$ the ellipticity of the term is sometimes not recognized by the numerical approximation of the equations. This short coming of the formulation prompted a search for a new formulation that could both use the concept of solvent viscosity and preserve the ellipticity of the momentum equation, while still allowing for ease of implementation of new models. Hence, the Elastic-Viscous Stress Splitting, EVSS, formulation was created. As the name suggests, the stress in the momentum equation is split into elastic and viscous components by defining the elastic stress as

$$\boldsymbol{\Sigma} = \boldsymbol{\tau} + \dot{\boldsymbol{\gamma}} = \boldsymbol{\tau}_p + (1-\beta)\,\dot{\boldsymbol{\gamma}} \tag{3.1}$$

Substituting this equation into the momentum equation, eq. 2.13, yields

$$-\nabla^2 \mathbf{v} + \nabla \cdot \boldsymbol{\Sigma} + \nabla p = \mathbf{0} \tag{3.2}$$

$$\nabla \cdot \mathbf{v} = 0 \tag{3.3}$$

$$\boldsymbol{\Sigma} + \mathrm{De}\boldsymbol{\Sigma}_{(1)} - \mathrm{De}\,(1-\beta)\,\dot{\boldsymbol{\gamma}}_{(1)} = \mathbf{0} \tag{3.4}$$

where the dependent variables are ($\mathbf{v}$, p, $\boldsymbol{\Sigma}$). The elliptic term in the momentum equation is guaranteed to survive for all choices of flow conditions and material parameters. However, introduction of the upper convected derivative of the rate-of-strain tensor proves difficult to deal with due to the time derivative term it introduces. Mesh sized oscillations were eliminated through the use of a least squares approximation of the rate-of-strain tensor,

$$\dot{\boldsymbol{\gamma}} - (\nabla \mathbf{v}) - (\nabla \mathbf{v})^T = 0 \tag{3.5}$$

However, numerical instabilities arise with use of this formulation at relatively low Deborah numbers for the planar Couette flow problem as shown by Szady [81].

Based on the knowledge that the constitutive equation formulated with EVSS degenerates into an algebraic equation along the curves where the velocity field vanishes, Brown [14] and Szady [82] changed the interpolated quantity from the rate-of-strain tensor to $\mathbf{G}$, the velocity gradient interpolant,

$$\mathbf{G} - \nabla \mathbf{v} = \mathbf{0} \tag{3.6}$$

This new definition of the rate-of-strain tensor in terms of the interpolant, $\dot{\boldsymbol{\gamma}} = (\nabla \mathbf{v}) + (\nabla \mathbf{v})^T = \mathbf{G} + \mathbf{G}^T$,is substituted into the constitutive equation. As demonstrated by Szady [81], the EVSS-G formulation for the momentum equation and Oldroyd-B constitutive equation shows no numerical instabilities for the planar Couette flow problem with Deborah numbers an order of magnitude larger than those where the EVSS formulation shows numerical instabilities to occur.

The Discrete Elastic-Viscous Split Stress Gradient interpolant formulation, DEVSS-G, is similar to the EVSS-G formulation, with a change in the elastic stress expression

$$\boldsymbol{\Sigma} = \boldsymbol{\tau} + \left[\mathbf{G} + \mathbf{G}^T\right] = \boldsymbol{\tau}_p + (1 - \beta)\left[\mathbf{G} + \mathbf{G}^T\right]$$

The elastic stress is substituted into the momentum equation only, yielding

$$\boldsymbol{\tau}_p + \mathrm{De}\boldsymbol{\tau}_{p(1)} + (1-\beta)\left[\mathbf{G} + \mathbf{G}^T\right] = \mathbf{0} \tag{3.7}$$

$$-\nabla^2\mathbf{v} + (1-\beta)\nabla\cdot\left[\mathbf{G} + \mathbf{G}^T\right] + \nabla\cdot\boldsymbol{\tau}_p + \nabla p = \mathbf{0} \tag{3.8}$$

$$\mathbf{G} - \nabla\mathbf{v} = \mathbf{0} \tag{3.9}$$

$$\nabla\cdot\mathbf{v} = 0 \tag{3.10}$$

Notice here that there is no longer an occurrence of the upper convected derivative of the rate-of-strain tensor in the constitutive equation. Thus, the difficulty of interpolating the time derivative of the rate-of-strain tensor is removed completely. The method also offers advantages in ease of derivation for different constitutive equations as compared to the EVSS and EVSS-G methods. Szady [81] demonstrates for the planar Couette flow that the EVSS-G and DEVSS-G formulations produce nearly identical results at high Deborah number with no introduction of numerical instabilities from the change in formulation. The DEVSS-G formulation represents the state of the art in the calculation of viscoelastic flows and is used exclusively in the work contained in this document.

3.2 Finite Element Method

3.2.1 Development of the finite element method

Since the finite element method is at the heart of the numerical methods used in this thesis, application of the method is illustrated here. The diffusion equation is chosen since it is one of the typical examples used in finite element texts and its form is very similar to that of the momentum equation. The following is just a condensed version of

the treatment found in many finite element texts and is included here for completeness. The reader is directed to [36], [37], and [42] for more detailed discussion.

Consider the two-point boundary value problem of the diffusion equation for a function $u(x)$ is defined as

$$-\frac{d^2u}{dx^2} = f \tag{3.11}$$

with a Dirichlet boundary condition applied at $x = 0$,

$$u(0) = 0 \tag{3.12}$$

and a Neumann boundary condition applied at $x = 1$,

$$\left.\frac{du}{dx}\right|_{x=1} = 0 \tag{3.13}$$

This form of the diffusion equation is termed the classical form or strong form.

An equivalent formulation of this system is to find a function $u \in V$ such that

$$a(u, v) = (f, v) \qquad \forall\ v \in V \tag{3.14}$$

where $a(.,.)$ and $(.,.)$ are vector valued functions defined as

$$a(w, v) \equiv \int_0^1 \left(\frac{dw}{dx}\right)\left(\frac{dv}{dx}\right) dx \tag{3.15}$$

$$(w, v) \equiv \int_0^1 wv dx \tag{3.16}$$

and the function space V is defined as

$$V = \left\{v : v \in C^0 \text{ on } [0,1],\ v(0) = 0\right\} \tag{3.17}$$

Proof of the equivalence of the strong and weak form of the diffusion equation can be

shown by performing integration by parts on eq. 3.14,

$$a(u,v) = -\int_0^1 v\frac{d^2u}{dx^2}dx + v\left.\frac{du}{dx}\right|_0^1 = \int_0^1 fvdx = (f,v) \tag{3.18}$$

which can be rearranged to arrive at

$$-\int_0^1 v\left(\frac{d^2u}{dx^2} + f\right)dx + v\left.\frac{du}{dx}\right|_0^1 = 0 \tag{3.19}$$

Notice here that if u is found such that eq. 3.11 is satisfied, then the integrand of the first term of eq. 3.19 will be zero. The requirement that $v \in V$ where V is constructed such that it vanishes at $x = 0$ causes the second term of eq. 3.19 to be zero at $x = 0$. Also, satisfying the Neumann boundary condition, eq. 3.13, will cause the second term of eq. 3.19 at $x = 1$ to be zero. Thus, satisfying eq. 3.19 is equivalent to solving the diffusion equation and boundary condtions prescribed by eqs. 3.11 - 3.13.

Systems of equations discretized using the finite element method are first placed in the weak form as shown above. The weak form of any differential equation can be constructed by multiplying the equation by a test function v and then integrating over the spatial domain Ω. Consideration of the boundary conditions is handled according to the specific type of condition. For Dirichlet boundary conditions like that given in eq. 3.12, an appropriate function space V is constructed only of functions that will vanish on the appropriate boundary. These boundary conditions are referred to as essential conditions. For Neumann boundary conditions, integration by parts is applied to the weak form equation to produce an additional term in the weak form as in eq. 3.19. These boundary conditions are referred to as natural conditions.

Discretization of the weak form of the boundary value problem is accomplished by applying the Ritz-Galerkin approximation, where the infinite dimensional function space V is replaced by a finite dimensional subspace $V_h \subset V$. The approximate variational

problem then becomes to find $u_h \in V_h$ such that

$$a(u_h, v) = (f, v) \qquad \forall\, v_h \in V_h \tag{3.20}$$

The functional space V_h is described by a linearly independent set of basis functions $\{\phi_i\}$ such that the functions in the space V_h can be expressed as $v(x) = \sum_i c_i \phi_i(x)$. By applying this discrete form of $v(x)$, eq. 3.20 is transformed to a set of linear equations,

$$\mathbf{A} \cdot \mathbf{u}_h = \boldsymbol{f} \tag{3.21}$$

where $\mathbf{A}$ is a two-dimensional tensor whose components are described by $A_{ij} \equiv a(\phi_i, \phi_j)$, $\mathbf{u}_h$ is the vector of coefficients u_{hi} in $u_h(x) = \sum_i u_{hi}\phi_i(x)$, and $\boldsymbol{f}$ is a vector whose components are described by $f_i = (f_i, v)$.

A number of different variational methods are defined by the choice of functional space V_h used to construct the linear set of equations in eq. 3.21 from eq. 3.20. The finite element method defines the functional space V_h by dividing the physical domain Ω into a finite number of non-overlapping subintervals I_j, or elements. V_h is then constructed by using low order, usually first and second order, polynomials defined on each of the elements. The piecewise linear approximation for the V_h space is defined as

$$V_h = \left\{ v : v \in C^0(\Omega),\ \ v|_{I_j} \in \Pi_1,\ v(0) = 0 \right\} \tag{3.22}$$

where C^0 is a continuous function and Π_n is the set of all real polynomials P with degree less than or equal to n,

$$P_n(x) = a_0 + a_1 x + ... + a_n x^n \tag{3.23}$$

The piecewise quadratic approximation for the V_h space is defined as

$$V_h = \left\{ v : v \in C^0(\Omega),\ \ v|_{I_j} \in \Pi_2,\ v(0) = 0 \right\} \tag{3.24}$$

In the viscoelastic flow problem, the polynomial approximations for the pressure, velocity, and velocity gradient interpolant unknowns are required to be continuous on the element boundaries. In addition to continuity between elements, the funtional spaces for velocity and pressure in the Stokes problem must meet the requirements given by the inf-sup condition, also known as the Ladyshenskaya-Babuška-Brezzi condition, to gaurantee compatability and uniqueness of the solution. The reader is directed to [37] for further discussion of the inf-sup condition. To this end, the function space given in eq. 3.22 is used for the pressure and velocity gradient interpolant unknowns, and that given in eq. 3.24 is used for the velocity unknowns.

Unlike the velocity, pressure and velocity gradient interpolant spaces, the space used to approximate the stress unknowns is not required to be continuous. In the Discontinuous Galerkin formulation, the functional space for stress is constructed from piecewise linear functions that are not required to be continuous on the elemental boundaries. Description of the this finite element space is aided by considering a typical stress evolution equation,

$$\frac{\partial \boldsymbol{\tau}}{\partial t} + \mathbf{v} \cdot \nabla \boldsymbol{\tau} = f(\boldsymbol{\tau}, \mathbf{G}) \tag{3.25}$$

The Discontinuous Galerkin space is then defined as

$$V_h^{DG} = \left\{ v : v \in L^1(\Omega), \;\; v|_{I_j} \in \Pi_1, \; v(0) = 0 \right\} \tag{3.26}$$

where L^1 is a continuous function within each element. Given this space, the weak for eq. 3.25 becomes to find $\boldsymbol{\tau}_h \in V_h^{DG}$ such that

$$\left(v, \frac{\partial \boldsymbol{\tau}_h}{\partial t} \right) + (v, \mathbf{v}_h \cdot \nabla \boldsymbol{\tau}_h) - \left(v, \mathbf{v}_h \cdot \mathbf{n} \left(\boldsymbol{\tau}_h^+ - \boldsymbol{\tau}_h^- \right) \right)_{\partial T_e^-} = (v, f(\boldsymbol{\tau}_h, \mathbf{G}_h)) \qquad \forall \; v \in V_h^{DG} \tag{3.27}$$

where $\mathbf{n}$ is the outward-pointing unit normal on the element surface and ∂T_e^- is an inflow

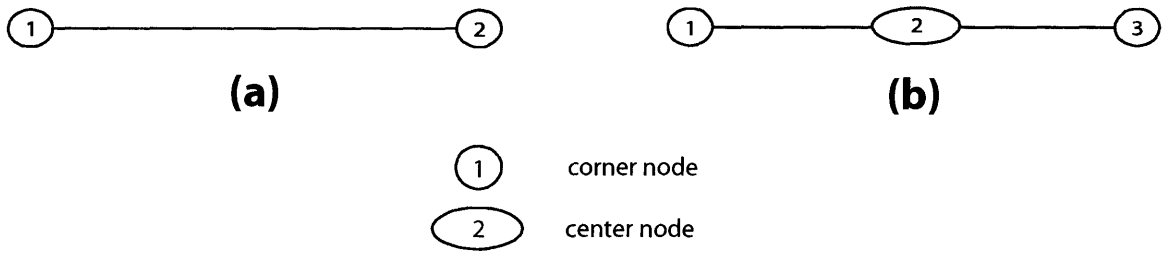


Figure 3-1: Line elements used to discretize one-dimensional geometries. (a): 2-node linear line element; (b): 3-node quadratic line element.

boundary of the element, defined as

$$\partial T_e^- \equiv \{x \in \partial T_e : (\mathbf{v} \cdot \mathbf{n}) \leq 0\} \tag{3.28}$$

Further discussion of the Discontinuous Galerkin method can be found in [23].

3.2.2 Elements and Basis Functions

Detailed discussion of many of the elements and basis functions used in the Finite Element Method can be found in [42]. The elements and basis functions used in this work can either be found there explicitly or derived from relationships contained within the text. For completeness, the elements and basis functions implemented in the two-dimensional finite element method for computations contained within this document are derived below.

Line elements are used to discretize a one-dimensional physical geometry, Fig. 3-1. Langrange polynomials, eq. 3.29, are used in the derivation of the basis functions for the line elements shown above.

$$l_a^{n_{en}-1}(\xi) = \frac{\prod_{\substack{b=1 \\ b \neq a}}^{n_{en}} (\xi - \xi_b)}{\prod_{\substack{b=1 \\ b \neq a}}^{n_{en}} (\xi_a - \xi_b)} \tag{3.29}$$

where n_{en} is the number of nodes in the one-dimensional element, $n_{en}-1$ is the order of the polynomial, a is the node in the element, and ξ_a is the coordinate of node a in ξ-space. Basis functions for an n_{en}-noded element in one dimension are defined by the relation

$$N_a(\xi) = l_a^{n_{en}-1}(\xi) \tag{3.30}$$

The basis functions for the linear (2-node) line element are given by

$$N_1(\xi) = l_1^1(\xi) = \frac{1}{2}(1-\xi) \tag{3.31a}$$

$$N_2(\xi) = l_2^1(\xi) = \frac{1}{2}(1+\xi) \tag{3.31b}$$

Similarly, the basis functions for the quadratic (3-node) line element are given by

$$N_1(\xi) = l_1^2(\xi) = \frac{1}{2}\xi(\xi-1) \tag{3.32a}$$

$$N_2(\xi) = l_2^2(\xi) = 1-\xi^2 \tag{3.32b}$$

$$N_3(\xi) = l_3^2(\xi) = \frac{1}{2}\xi(\xi+1) \tag{3.32c}$$

The choice of elements in the two-dimensional space is somewhat more complicated. A quadrilateral element can be derived as a simple extension of the one-dimensional line element, Fig. 3-2. Basis functions of the two-dimensional elements are found by taking the product of two one-dimensional Lagrange polynomials

$$N_a(\xi,\eta) = l_b^{n_{en}-1}(\xi)\, l_c^{n_{en}-1}(\eta) \tag{3.33}$$

where b and c can be thought of as the nodes of two separate one-dimensional elements, one oriented along the ξ axis, the other along the η axis, and a refers to the node on the two-dimensional element. Basis functions for the two-dimensional bilinear quadrilateral

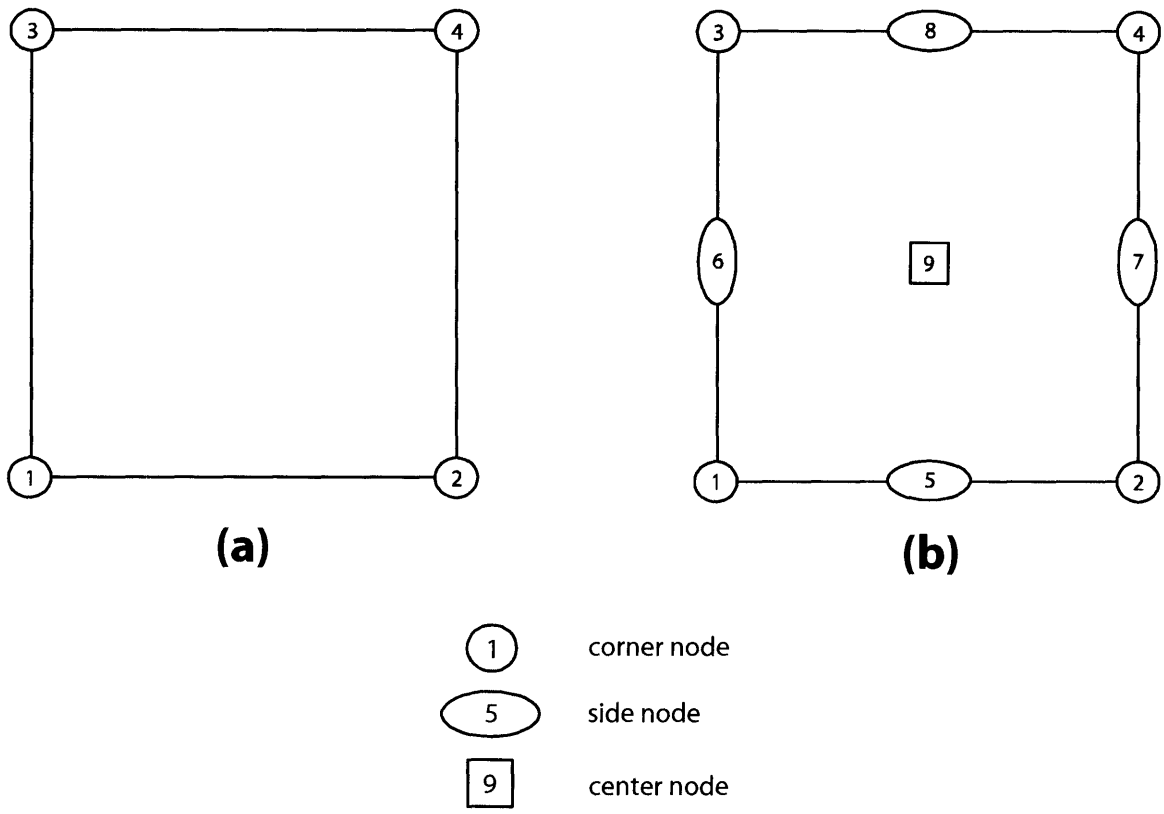


Figure 3-2: Quadrilateral elements used to discretize the surfaces of a three-dimensional geometry. (a): 4-node bilinear quadrilateral element; (b): 9-node biquadratic quadrilateral element.

element are then

$$N_1(\xi,\eta) = l_1^1(\xi)\, l_1^1(\eta) = \frac{1}{4}(1-\xi)(1-\eta) \tag{3.34a}$$

$$N_2(\xi,\eta) = l_2^1(\xi)\, l_1^1(\eta) = \frac{1}{4}(1+\xi)(1-\eta) \tag{3.34b}$$

$$N_3(\xi,\eta) = l_1^1(\xi)\, l_2^1(\eta) = \frac{1}{4}(1-\xi)(1+\eta) \tag{3.34c}$$

$$N_4(\xi,\eta) = l_2^1(\xi)\, l_2^1(\eta) = \frac{1}{4}(1+\xi)(1+\eta) \tag{3.34d}$$

Similarly, basis functions for the two-dimensional biquadratic quadrilateral element are then

$$N_1(\xi,\eta) = l_1^2(\xi)\, l_1^2(\eta) = \frac{1}{4}\xi\eta(\xi-1)(\eta-1) \tag{3.35a}$$

$$N_2(\xi,\eta) = l_3^2(\xi)\, l_1^2(\eta) = \frac{1}{4}\xi\eta(\xi+1)(\eta-1) \tag{3.35b}$$

$$N_3(\xi,\eta) = l_1^2(\xi)\, l_3^2(\eta) = \frac{1}{4}\xi\eta(\xi-1)(\eta-1) \tag{3.35c}$$

$$N_4(\xi,\eta) = l_3^2(\xi)\, l_3^2(\eta) = \frac{1}{4}\xi\eta(\xi+1)(\eta+1) \tag{3.35d}$$

$$N_5(\xi,\eta) = l_2^2(\xi)\, l_1^2(\eta) = \frac{1}{2}\eta\left(1-\xi^2\right)(\eta-1) \tag{3.35e}$$

$$N_6(\xi,\eta) = l_1^2(\xi)\, l_2^2(\eta) = \frac{1}{2}\xi(\xi-1)\left(1-\eta^2\right) \tag{3.35f}$$

$$N_7(\xi,\eta) = l_3^2(\xi)\, l_2^2(\eta) = \frac{1}{2}\xi(\xi+1)\left(1-\eta^2\right) \tag{3.35g}$$

$$N_8(\xi,\eta) = l_2^2(\xi)\, l_3^2(\eta) = \frac{1}{2}\eta\left(1-\xi^2\right)(\eta+1) \tag{3.35h}$$

$$N_9(\xi,\eta) = l_2^2(\xi)\, l_2^2(\eta) = \left(1-\xi^2\right)\left(1-\eta^2\right) \tag{3.35i}$$

Triangular elements can also be derived for the two-dimensional discretization, Fig. 3-3. They can serve one of two purposes, either as the sole type of element for discretization of the entire physical domain or as a transition element between regions of differing degrees of refinement of quadrilateral elements. Triangular coordinates, r, s, and $t(r,s) = 1-r-s$, are used to define the basis functions for these elements, in which each coordinate is 0 along one edge of the triangle and 1 at the opposite node. The

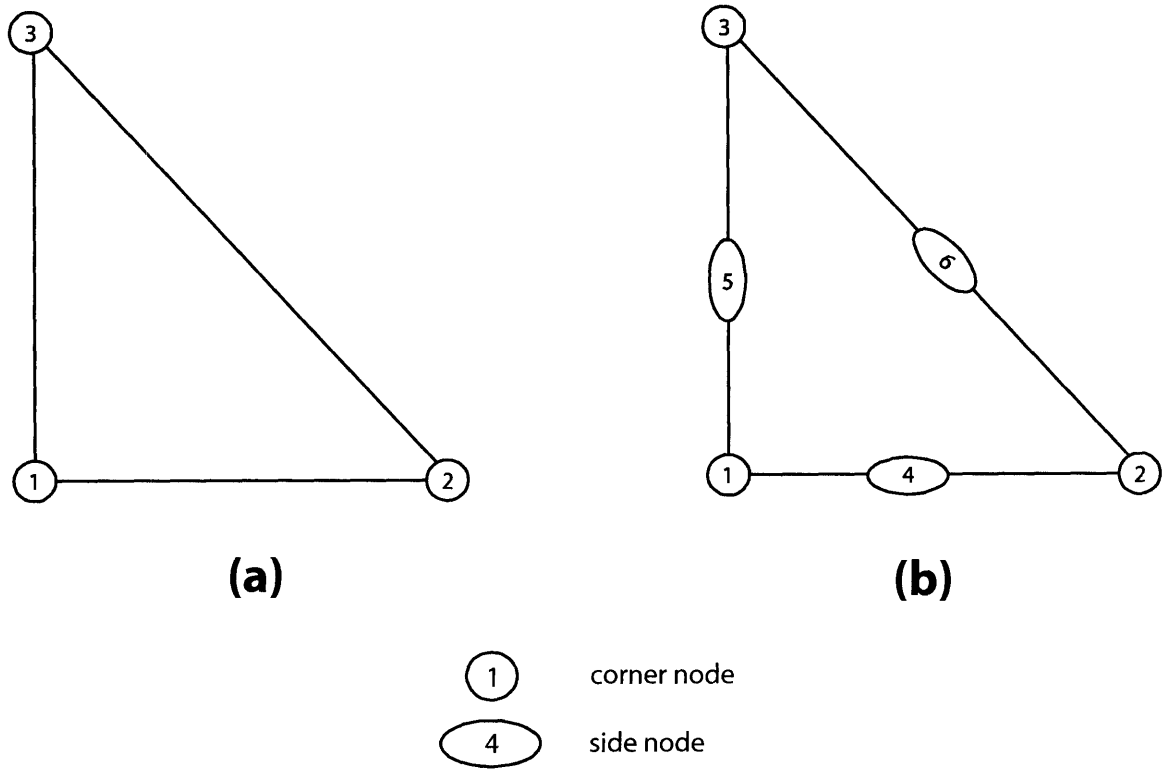


Figure 3-3: Triangle elements used to discretize the surfaces of a three-dimensional geometry. (a): 3-node bilinear triangle element; (b): 6-node biquadratic triangle element.

general formula for the derivation of basis functions for the Lagrange-type triangular elements is

$$N_a(r,s,t) = T_I(r)\,T_J(s)\,T_K(t) \tag{3.36}$$

where

$$T_I(r) = \begin{cases} l_I^{I-1}\left(\frac{2r}{r_I-1}\right), & I \neq 1 \\ 1, & I = 1 \end{cases} \tag{3.37}$$

The basis functions for the bilinear triangular element are then

$$N_1(r,s,t) = T_2(r)\,T_1(s)\,T_1(t) = l_2^1\left(\frac{2r}{r_2-1}\right) = r \tag{3.38a}$$

$$N_2(r,s,t) = T_1(r)\,T_2(s)\,T_1(t) = l_2^1\left(\frac{2s}{s_2-1}\right) = s \tag{3.38b}$$

$$N_3(r,s,t) = T_1(r)\,T_1(s)\,T_2(t) = l_2^1\left(\frac{2t}{t_2-1}\right) = t \tag{3.38c}$$

Similarly, the basis functions for the biquadratic triangular element are

$$N_1(r,s,t) = T_3(r)\,T_1(s)\,T_1(t) = l_3^2\left(\frac{2r}{r_3-1}\right) = r(2r-1) \tag{3.39a}$$

$$N_2(r,s,t) = T_1(r)\,T_3(s)\,T_1(t) = l_3^2\left(\frac{2s}{s_3-1}\right) = s(2s-1) \tag{3.39b}$$

$$N_3(r,s,t) = T_1(r)\,T_1(s)\,T_3(t) = l_3^2\left(\frac{2t}{t_3-1}\right) = t(2t-1) \tag{3.39c}$$

$$N_4(r,s,t) = T_2(r)\,T_2(s)\,T_1(t) = l_2^1\left(\frac{2r}{r_2-1}\right) l_2^1\left(\frac{2s}{s_2-1}\right) = 4rs \tag{3.39d}$$

$$N_5(r,s,t) = T_1(r)\,T_2(s)\,T_2(t) = l_2^1\left(\frac{2s}{s_2-1}\right) l_2^1\left(\frac{2t}{t_2-1}\right) = 4st \tag{3.39e}$$

$$N_6(r,s,t) = T_2(r)\,T_1(s)\,T_2(t) = l_2^1\left(\frac{2r}{r_2-1}\right) l_2^1\left(\frac{2t}{t_2-1}\right) = 4rt \tag{3.39f}$$

The work in this document uses quadrilateral elements as the primary element for discretization of the physical geometries and relies on triangular elements only as transition elements between regions of differing degrees of refinement.

3.3 Decoupled sub-problem formulation

Two major classes of numerical methods used for finding the steady-state solutions of systems of partial differential equations are the direct computation of the steady-state solution through a fully coupled set of equations and indirect computation through a decoupled set of equations using time-dependent methods to evolve the problem to steady-state. Direct methods for computing the steady-state solution involve solving large systems of equations using Newton's method or some variant. This involves formation of a large Jacobian matrix of derivatives of all discretized equations with respect to all of the system unknowns. Factorization of this matrix can be carried out by using sparse LU factorization or some other efficient numerical method, but for larger problems frontal solvers and even iterative solvers must be employed. The major advantages of the method are guaranteed quadratic convergence and the availability of a Jacobian for analysis of bifurcations. One of the major disadvantage of this technique is the need for a good initial guess for the solver. While the method is guaranteed to converge given a good initial guess, one is not always readily available.

Time-dependent methods are often used because of the robustness added over the steady-state methods of not needing a good initial guess in order to find the steady-state solution of a system. By starting out with any initial guess, the problem solution is time-stepped toward steady state. Limitations exist on how large the time step can be, dictated by the stability of the system and the time integration algorithm utilitzed. Apart from the robustness of the method to the initial guess, other more important changes can be made to help with the size of the computational problem. Since the entire set of equations no longer must be solved simultaneously, the evolution equations describing the polymer stress can be decoupled from the equations describing flow and solved for in separate substeps in time. This approach is known as operator splitting [75]. An illustration of the decoupled set of equations is shown in Fig. 3-4.

Two major advantages can be identified with the operator splitting approach. By decoupling the stress equations from the momentum and mass equations, different types

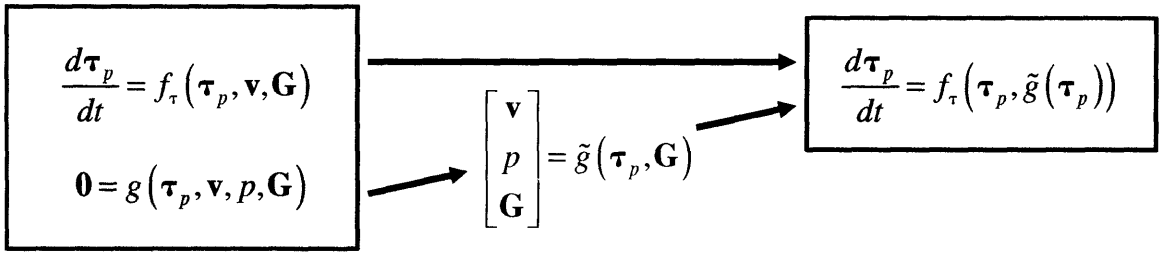


Figure 3-4: Illustration of the index one set of differential algebraic equations for the viscoelastic flow problems. The set of DAE's can be rewritten as a set of ODE's describing the stress evolution which are constrained by the flow equations.

of constitutive equations can be implemented more easily. For example, given the flow field, equations describing the evolution of molecular properties of the fluids can be used with only knowledge of how stress can be expressed in terms of them. No knowledge of direct effects on velocity and pressure is needed. This allows for the use of a much broader range of constitutive equations to describe the dynamics of the system based on molecular effects, etc..

Another major advantage of being able to separate the momentum and mass equations from the constitutive equations is the reduction of overall problem size. As an illustration, consider solution of a problem of n total unknowns, consisting of two subproblem of $n*1/2$ unknowns each. Using an iterative solver for the fully coupled set of equations requires order n^2 operations to solve the set. By decoupling the equations and splitting them into the two subproblems, we expect order $(n*1/2)^2 = 1/4*n^2$ operations to reach solution for each subproblem and $1/2*n^2$ operations for the total problem, half of the operations needed as compared to the fully coupled problem. From this simple illustration it is apparent that a large advantage in the size of tractable problems is possible by applying the operator splitting to the set of equations describing the viscoelastic fluid.

3.4 Time-Stepping Algorithms

As discussed above, integration of the time-dependent equations to reach steady state offers advantages over direct calculation of the steady state. There are a number of different time-stepping algorithms available to discretize the viscoelastic equations in time. This section contains the most relavant methods. For a more complete treatment of the subject, the reader is directed to [62] and [39].

3.4.1 Taylor-series-based methods

The most basic class of the time-stepping algorithms is the set of explicit methods. These methods require only knowledge of the state of the system at the previous time-step in order to construct the state at the current time-step. The simplest form of this class of methods is Euler's method, also known as the Forward Euler method. The Taylor series expansion of y in time around t is given by,

$$y\left(t+\Delta t\right)=y\left(t\right)+\frac{dy\left(t\right)}{dt}\Delta t+\frac{d^{2}y\left(t\right)}{dt^{2}}\Delta t^{2}+\cdots \tag{3.40}$$

Euler's method is derived by dropping the terms of second order and higher leaving

$$y_{k+1}=y_{k}+\left.\frac{dy}{dt}\right|_{k}\Delta t \tag{3.41}$$

where the shorthand y_k is used to represents the solution $y\left(t_k\right)$ and y_{k+1} represents $y\left(t_k+\Delta t\right)$. The method is explicit, since it requires only information at the current timestep to construct the solution at the next timestep. The method is said to be first-order accurate, since the terms of second order and higher are neglected. The clear advantage of Euler's method is that it requires only the current value of y and the derivative of y with respect to time t at the current time to construct the solution at time $t+\Delta t$. Both of these quantities are typically readily available. The disadvantage of this method is that it tends to have poor numerical stability irrespective of the stability

of the equation set; it therefore requires a relatively small timestep to remain stable.

The implicit method analogous to the Forward Euler method is the Backward Euler method. The Backward Euler method is derived in the same manner as the Forward Euler method, except that the first derivative of y with respect to time is calculated at the future timestep rather than the current timestep,

$$y_{k+1} = y_k + \left.\frac{dy}{dt}\right|_{k+1} \Delta t \tag{3.42}$$

Unlike the Forward Euler method, the Backward Euler method is unconditionally stable. However, like Forward Euler method, the Backward Euler method is only first order accurate, severly limiting its usefulness. Methods of higher order accuracy are typically more desirable.

3.4.2 Runge-Kutta methods

To decrease the computational time needed to reach a steady-state solution, it is of interest to use the largest time-step possible when time integrating the set of equations. The limit of the size of the stable time step can be increaesd by including higher order derivatives with respect to time from the Taylor series. Unfortunately these quantities are not often readily available and can be difficult to calculate in more complex equation sets. Other methods have been developed to increase the stable time step without the need to evaluate the higher order terms. One of the most well-known sets of methods of this type is the Runge-Kutta method set. These methods simulate the higher order terms by evaluating the time derivative of y at several steps in between k and $k+1$.

The most often used of these methods is the 4th order Runge-Kutta method, RK4,

$$y_{k+1} = y_k + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{3.43a}$$

$$k_1 = f(t_k, y_k)\,\Delta t \tag{3.43b}$$

$$k_2 = f(t_k + \Delta t/2, y_k + k_1/2)\,\Delta t \tag{3.43c}$$

$$k_3 = f(t_k + \Delta t/2, y_k + k_2/2)\,\Delta t \tag{3.43d}$$

$$k_4 = f(t_k + \Delta t, y_k + k_3)\,\Delta t \tag{3.43e}$$

where $f(t_k, y_k) = \frac{dy_k}{dt}\big|_k$ is the derivative of y_k with respect to t at time t_k. While the method does not have any formal estimate of error to base a stable time-step on, application of the method shows a significant increase in the stable time-step compared to the lower order methods [62]. The RK4 method is used exclusively in the integration of the evolution equations for stress in the calculations presented in this work.

3.5 Parallel solution method

Aside from improvements in the formulation of robust, efficient numerical methods to compute solutions of the partial differential equations describing the dynamics of viscoelastic fluid systems, strides have also been made in employing parallel architectures to further increase the total tractable problem size. Caola [16] describes in detail one such method and demonstrates the use of the method on complex flow systems of viscoelastic fluids. The method is summarized here for completeness.

The method uses the time-dependent DEVSS-G/DG formulation in the decoupled form discussed in Section 3.3. By attacking the set of equations describing flow and that describing polymer stress separately, optimal numerical techniques can be applied to each set. The set of equations can be written in the form

$$\mathbf{M} \cdot \frac{d\boldsymbol{\tau}_p}{dt} = \frac{1}{\mathrm{De}} \boldsymbol{f}(\boldsymbol{\tau}_p, \mathbf{v}, \mathbf{G}) \tag{3.44}$$

$$\boldsymbol{g}\left(\boldsymbol{\tau}_p, \mathbf{v}, p, \mathbf{G}\right) = \mathbf{0} \tag{3.45}$$

where the evolution equations for stress and the equations describing the flow are separated and where $\boldsymbol{f}$ represents the discretization of the constitutive equation and $\boldsymbol{g}$ represents the discretization of the generalized Stokes problem. Since the two sets of equation are decoupled from one another, eq. 3.45 can be written as the linear set of equations

$$\begin{bmatrix} \mathbf{v} \\ p \\ \mathbf{G} \end{bmatrix} = \boldsymbol{g}\left(\boldsymbol{\tau}_p\right) \tag{3.46}$$

Eq. 3.46 is then substituted into eq. 3.44 to yield

$$\hat{\boldsymbol{f}}\left(\boldsymbol{\tau}_p\right) \equiv \mathbf{M}^{-1}\boldsymbol{f}\left(\boldsymbol{\tau}_p, \boldsymbol{g}\left(\boldsymbol{\tau}_p\right)\right) \tag{3.47}$$

The fourth order Runge-Kutta method, eq. 3.43a, is applied to the eqs. 3.44 and 3.46 to yield

$$k_1 = \Delta t \hat{\boldsymbol{f}}\left(\boldsymbol{\tau}_p^n\right) \tag{3.48a}$$

$$k_2 = \Delta t \hat{\boldsymbol{f}}\left(\boldsymbol{\tau}_p^n + \frac{k_1}{2}\right) \tag{3.48b}$$

$$k_3 = \Delta t \hat{\boldsymbol{f}}\left(\boldsymbol{\tau}_p^n + \frac{k_2}{2}\right) \tag{3.48c}$$

$$k_4 = \Delta t \hat{\boldsymbol{f}}\left(\boldsymbol{\tau}_p^n + k_3\right) \tag{3.48d}$$

$$\boldsymbol{\tau}_p^{n+1} = \boldsymbol{\tau}_p^n + \frac{1}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right) \tag{3.48e}$$

Evaluation for each of the k_i substeps requires the evaluation of the generalized Stokes problem given in eq. 3.46. Because of the discretization of the stress unknowns with the discontinuous Galerkin basis functions, $\mathbf{M}$ in eq. 3.47 for each element in the domain is independent of all other elements. Parallelization of the evaluation of stress from eqs. 3.48a-e is then accomplished by distributing the elements among the processors in the

parallel machine. Since the elemental equations are independent from one another, the stress equations are ideally parallelizable, in that doubling the number of processors in the parallel machine will result in one half the time needed to reach solution.

Solution of the generalized Stokes problem given in eq. 3.46 is accomplished by preconditioning the set of equations with the BCALM preconditioner [15] and solving them with the GMRES Krylov subspace iterative method. Details of the GMRES method and other Krylov subspace methods can be found in [66]. Discretization of the generalized Stokes problem yields the linear system

$$\hat{\mathbf{A}}\mathbf{x} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \end{bmatrix} = \mathbf{b} \tag{3.49}$$

where $\mathbf{A}$ and $\mathbf{B}$ are the discretizations of eqs. 3.8 and 3.9, $\mathbf{C}$ and $\mathbf{D}$ are the discretizations of eq. 3.10, and $\mathbf{x}_1$ and $\mathbf{x}_2$ are vectors of the sets of unknowns $[\mathbf{v}, \mathbf{G}]$ and $[p]$, respectively. Note that $\mathbf{D} = \mathbf{0}$ results from the discretization, since pressure does not appear in eq. 3.10. $\hat{\mathbf{A}}$ can then be rewritten in the Schur complement form as

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0} & \mathbf{S} \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 - \mathbf{C}\mathbf{A}^{-1}\mathbf{b}_1 \end{bmatrix} \tag{3.50}$$

where the Schur complement $\mathbf{S}$ is

$$\mathbf{S} \equiv \mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B} \tag{3.51}$$

Eq. 3.50 can now be solved in two steps,

$$\mathbf{S}\mathbf{x}_2 = \left(\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}\right)\mathbf{x}_2 = \mathbf{b}_2 - \mathbf{C}\mathbf{A}^{-1}\mathbf{b}_1 \tag{3.52a}$$

$$\mathbf{A}\mathbf{x}_1 = \mathbf{B}\mathbf{x}_2 + \mathbf{b}_1 \tag{3.52b}$$

where a separate preconditioner can now be applied to each step, detailed in [16] and [15]. $\mathbf{S}$ in eq. 3.52a is preconditioned by first approximating the Schur complement as $\mathbf{S} \cong \mathbf{D} - \mathbf{CB}$ and then preconditioning the resulting matrix with the diagonal of $\mathbf{S}$. Construction of a preconditioner for $\mathbf{A}$ in eq. 3.52b requires approximation of $\mathbf{A}^{-1}$ which is accomplished by using a two-level additive Schwarz domain decomposition method [69]. Level one of the method captures the high spatial frequency properties of the solution of $\mathbf{A}$, while the coarse mesh approximation in level two captures the low frequency properties of the solution. While the additive Schwarz method completely decouples the individual subdomains in the computational domain, the number of Krylov iterations needed to reach convergence can be greatly reduced by introducing overlap between the subdomains on the order of a single set of nodes [15].

Decomposition the geometrical domain into subdomains is accomplished through a routine called CHACO developed at Sandia National Lab in New Mexico. The decomposition minimizes the amount of surface area between the subdomains, reducing the amount of communication between processors required during solution of eq. 3.52b.

The end result is a method for which speed scales nearly linearly with the number of processors in the parallel machine, theoretically allowing computations of large problems limited only by the number of processors available. Eventually, a saturation in the number of processors occurs where adding additional processors no longer increases the speed of the calculation. This is because the ratio of communication between processors to computation per processor increases as the problem domain is broken up into smaller and smaller pieces. Intuitively, the saturation is delayed for larger and larger problem sizes as can be seen in Fig. 3-5. The speedup shown in Fig. 3-5 is a common measure of the performance of a parallel method given by

$$S \equiv \frac{t_1}{t_N} \tag{3.53}$$

where t_1 is the time required for solution of the problem on 1 processor and t_N is the time required for solution on N processors. The smallest mesh, SM1, has a total of 60,900

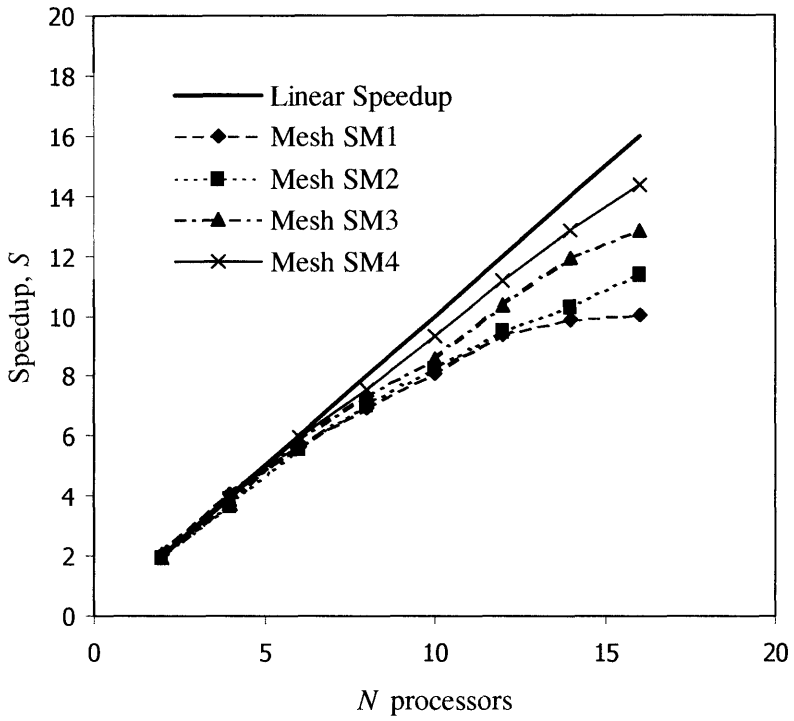


Figure 3-5: Speedup S for the solution of the Stokes-like linear system; De=0.5. The meshes range in size from mesh SM1 with 60,900 degrees of freedom to mesh SM4 with 751,110 degrees of freedom. Figure reproduced from [16]

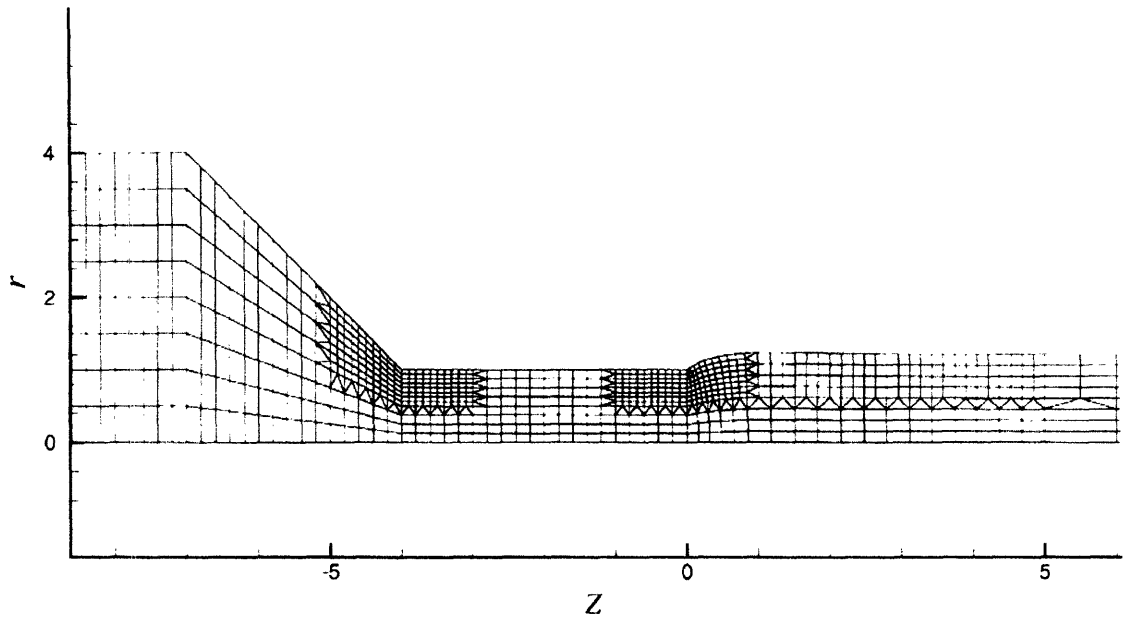


Figure 3-6: Finite element mesh used in simulation of the melt spinning process. The mesh contains 3896 elements and 92418 unknowns. Note that this unkonwn count is for continuous linear stress unkowns. Taken from [44].

degrees of freedom, and the largest mesh, SM4, has a total of 751,110 degrees of freedom. As can be seen from Fig. 3-5, the saturation level has not been reached at 16 processors for the SM4 mesh. The method has been used to compute problems of nearly 1,000,000 unknowns on a 32 processor parallel cluster.

3.6 Problem size estimates for 3-D geometries

To help the reader to understand the size of problems that are of interest for a full simulation of three-dimensional flow and thus the need for optimized numerical methods, an estimation of the three-dimensional problem size is presented here based on an adequately-resolved two-dimensional fiber spinning mesh found in the literature [44] and shown in Fig. 3-6. The two-dimensional mesh is expanded around the asmuthal direction

to construct the three-dimensional mesh. To estimate the number of elements in the three-dimensional mesh, a mesh that is uniform in the asmuthal direction will be used. Since the gradients of the flow and stress field should be the highest in the contraction region of the geometry between $z = -4$ and $z = 0$ in Fig. 3-6, the center of this region, at $r = 0.5$ will have elements of aspect ratio approximately 1:1:1. Elements in this region are of dimension $\Delta r = 0.25$ and $\Delta z = 0.2$ in the two-dimensional mesh. If there are 14 elements in the asmuthal direction, then elements in this region of the three-dimensional mesh will have dimensions of $\Delta r = 0.25$, $\Delta z = 0.2$, and $\Delta\theta = 0.225$. The total number of elements in the three-dimensional mesh for this fiber spinning geometry is then 54,544 with a total number of unknowns of approximately $N_{total} = 5,000,000$. Note that the estimate for the number of unknowns in the three-dimensional mesh takes into account the use of discontinuous basis functions used to represent the stress unknowns, as the three-dimensional finite element package makes use of this formulation. The total number of unknowns breaks down into roughly $N_{vp} = 1,670,000$ velocity and pressure unknowns, $N_{G\,\text{total}} = 830,000$ velocity gradient unknowns total with $N_G = 92,000$ for each of the nine components, and $N_s = 2,500,000$ stress unknowns. A typical bandwidth for a mesh of this size is $b = 2000$.

To estimate the time necessary to compute the solution to the above problem for a Deborah number of $O(1)$, solution methods using the steady-state method, the serial decoupled time-dependent method, and the parallel decoupled time-dependent method are considered. Assuming that the calculations are carried out on a computer capable of 100 million floating point operations per second (MFLOPS), the estimates for time to reach solution are as follows. For the steady-state method using a frontal solver to reduce the set of equations, the time to reach solution is found by

$$t_{\text{steady state}} = \frac{(N_{\text{total}} b^2) * n_{\text{steps}}}{100\text{x}10^6 \text{ FLOPS}} \tag{3.54}$$

where $n_{steps} = 100$ is the estimated number of Newton method steps needed to converge with consideration given for the need to take incremental steps in De to reach $O(1)$. The

amount of time is then

$$t_{\text{steady state}} = 38 \text{ years}$$

which is an infeasible amount of time to allot. For the serial decoupled time-dependent method using a preconditioned Krylov subspace method to solve the momentum/mass continuity and the velocity gradient interpolant subproblems and a 4th order Runge-Kutta time-stepping method to integrate the stress equations, the time to reach the steady-state solution is dominated by the time of the momentum/mass continuity subproblem. This time is given by

$$t_{\text{decoupled}} = \frac{(N_{\text{vp}}b) * t_{\text{steps}} * 4}{100\text{x}10^6 \text{ FLOPS}} \tag{3.55}$$

where $t_{steps} = 10000$ is the estimated number of time steps needed to reach steady state and the factor of 4 is for the four substeps in the 4th order Runge-Kutta method. The amount of time is then

$$t_{\text{decoupled}} = 15 \text{ days}$$

still somewhat longer than desired. Parallelization of the decoupled time-dependent method can offer significant savings in the time needed to compute a solution due to the division of labor among many processors in the parallel machine. For a good parallel algorithm, a linear relationship between the number of processors used in the parallel machine and the time needed to solve the problem is seen [16]. This translates to the time needed to compute the solution being given by

$$t_{\text{parallel}} = \frac{t_{\text{decoupled}}}{n_{\text{processors}}} \tag{3.56}$$

where $n_{processors}$ is the number of processors in the parallel machine. Assuming a parallel

machine of 32 processors, the time to compute solution for this method is

$$t_{\text{parallel}} = 0.5 \text{ days}$$

a very reasonable amount of time for a calculation of this size. Results of the estimated time to solution for the three methods are summarized in Table 3.1. Clearly it is desirable,

method	**time to reach solution**
steady	38 years
serial decoupled	15 days
parallel decoupled	0.5 days

Table 3.1: Estimated time to reach solution for the steady-state method, the serial decoupled time-dependent method, and the parallel decoupled time-dependent method.

and in the case of even larger problems necessary, to use a parallel version of the decoupled time-dependent formulation.

Chapter 4

Decoupled G Formulation

4.1 Introduction

The main goal of this chapter is to develop a new formulation for the solution of the viscoelastic flow equations adapted from the DEVSS-G formulation first used by Liu et al. [54]. Due to the relatively large computational burden of solving for the flow field when using continuum-based differential constitutive equations, it is desirable to minimize the overall size of the equation set used to describe the flow problem without loss of physical description. Optimizing the numerical methods used to translate the flow equations into a form solvable on the computer is the best way to attain this goal.

The following work details two modified forms of the DEVSS-G formulation in which the solution of the velocity gradient interpolant, $\mathbf{G}$, is decoupled from the solution of the momentum and mass continuity equations. The first method is a direct extension of the DEVSS-G method, utilizing a global least squares minimization to approximate $\mathbf{G}$. The second method employs a patch algorithm developed by Zienkiewicz and Zhu [90] to approximate $\mathbf{G}$ from local velocity gradient information. Comparison of the performance and accuracy of the two methods is made to determine that which best reduces the overall size of the flow problem. Comparisons to the accuracy of the fully coupled formulation are also presented.

4.2 Problem Description

The methods developed here are based on the DEVSS-G method used for solution of the equations describing viscoelastic flow. The DEVSS-G method is presented in full in Section 3.1. A decoupled form of the equations describing the polymer stress and the flow field is used as a starting point for the numerical method. This method is detailed in Section 3.3. The portion of the equation set and formulation that is of interest herein is the expression used to approximate the **G** interpolant and the method used to solve for that expression in relation to the momentum and mass continuity equations.

In order to reduce the overall size of the discretized flow equations, the equations approximating the **G** interpolant are decoupled from the flow equations. In the spirit of the decoupling of the polymer stress equations from the flow equations, where the solution of the flow equations acts as a constraint on the evolution equations for the polymer stress and precisely describes the flow field for the current stress field in an intertialess flow, the equations for the **G** interpolant are solved for after the velocity field has been determined for the current polymer stress in the system. The decoupled equations describing the polymer flow system take the form of eqs. 4.1, 4.2, and 4.3:

$$\frac{d\boldsymbol{\tau}_p}{dt} = \mathbf{M}^{-1} f\left(\boldsymbol{\tau}_p, g_1\left(\boldsymbol{\tau}_p, \mathbf{G}\right), g_2\left(\mathbf{v}\right)\right) \tag{4.1}$$

$$\begin{bmatrix} \mathbf{v} \\ p \end{bmatrix} = g_1\left(\boldsymbol{\tau}_p, \mathbf{G}\right) \tag{4.2}$$

$$\begin{bmatrix} \mathbf{G} \end{bmatrix} = g_2\left(\mathbf{v}\right) \tag{4.3}$$

A schematic diagram of the algorithm is shown in figure 4-1. In this form, eqs. 4.2 and 4.3 act as constraints on eq. 4.1. The following sections concern the details of the computation of eq. 4.3 for the global least squares minimization and the local smoothing methods. For both cases, only three components of **G** are solved for in the two-dimensional case (eight in the three-dimensional case): $\mathbf{G}_{xx}$, $\mathbf{G}_{yx}$, and $\mathbf{G}_{xy}$. The fourth component, $\mathbf{G}_{yy}$,

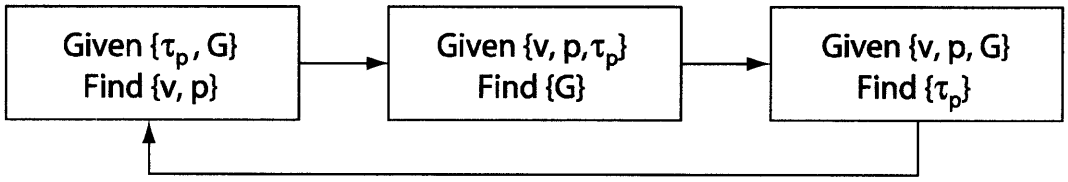


Figure 4-1: Time stepping algorithm of the decoupled **G** formulation for the equations describing viscoelastic flow.

is found using continuity, $\mathbf{G}_{xx} + \mathbf{G}_{yy} = 0$.

4.2.1 Global Least Squares G Formulation

The global minimization of the least squares approximation for the **G** interpolant is the form used in the DEVSS-G formulation detailed in Section 3.1 and shown in eq. 4.4:

$$\int_{\Omega} (\nabla \mathbf{v} - \mathbf{G})^2 \, d\Omega = 0 \tag{4.4}$$

Details on using least squares equations with the finite element method can be found in [43]. The least squares minimization then takes the variational form given in eq. 4.5 in the finite element discretization.

$$\int_{\Omega_e} (\nabla \mathbf{v} - \mathbf{G}) \, \psi d\Omega = 0 \tag{4.5}$$

Considered in the overall algorithm for solution of the polymer flow (fig. 4-1), the **G** interpolant is computed from eq. 4.4 given the velocity field at every linear node in the flow domain. For a given velocity field, the **G** interpolants are solved for at once for the entire flow domain. Expressing eq. 4.5 in component form (eq. 4.6),

$$\int_{\Omega_e} \left(\begin{bmatrix} \frac{dv_x}{dx} & \frac{dv_y}{dx} & \frac{dv_z}{dx} \\ \frac{dv_x}{dy} & \frac{dv_y}{dy} & \frac{dv_z}{dy} \\ \frac{dv_x}{dz} & \frac{dv_y}{dz} & \frac{dv_z}{dz} \end{bmatrix} - \begin{bmatrix} G_{xx} & G_{xy} & G_{xz} \\ G_{yx} & G_{yy} & G_{yz} \\ G_{zx} & G_{zy} & G_{zz} \end{bmatrix} \right) \psi d\Omega = 0 \tag{4.6}$$

and then considering the equation for each component (eq. 4.7),

$$\int_{\Omega_e} \left(\nabla v_{ij} - G_{ij}\right) \psi d\Omega = 0 \tag{4.7}$$

it is clear that each component of the **G** tensor is independent of one another; hence, each component can be solved for separately, reducing the size of the overall calculation.

4.2.2 Local Patch G Formulation

The concept of replacing global minimizations with local minimizations was first discussed by Hinton and Campbell [41]. Zienkiewicz and Zhu [90] furthered this concept by developing a patch recovery technique which allows for the computation of derivatives of the field variables through a local smoothing technique. Their method has the advantage over direct differentiation of preserving the order of the differentiated variable and the advantage over global minimization of greatly reducing the overall problem size by eliminating coupling of the unknowns at the global level. By using an adaptation of the patch algorithm laid out by Zienkiewicz and Zhu, the global minimization of **G** is replaced by local smoothing equations in which velocity gradient information from nearest node neighbors are used to compute a smoothed **G** interpolant (eq. 4.8).

$$\sum_{i=1}^{n} \mathbf{P}^T \left(x_i, y_i\right) \mathbf{P} \left(x_i, y_i\right) \mathbf{G}_{kl} = \sum_{i=1}^{n} \mathbf{P}^T \left(x_i, y_i\right) \nabla \mathbf{v}_{kl} \left(x_i, y_i\right) \tag{4.8}$$

Here n is the number of nodes in each element of the patch, $\nabla \mathbf{v}_{kl}$ is the k, l component of the velocity gradient computed from the flow field and $\mathbf{G}_{kl}$ is the k, l component of velocity gradient interpolant. The vector **P** is the polynomial expansion, or basis function, for an individual element in the patch. The polynomial expansion is given in eq. 4.9 as

$$\mathbf{P} = [1, x, y, xy] \tag{4.9}$$

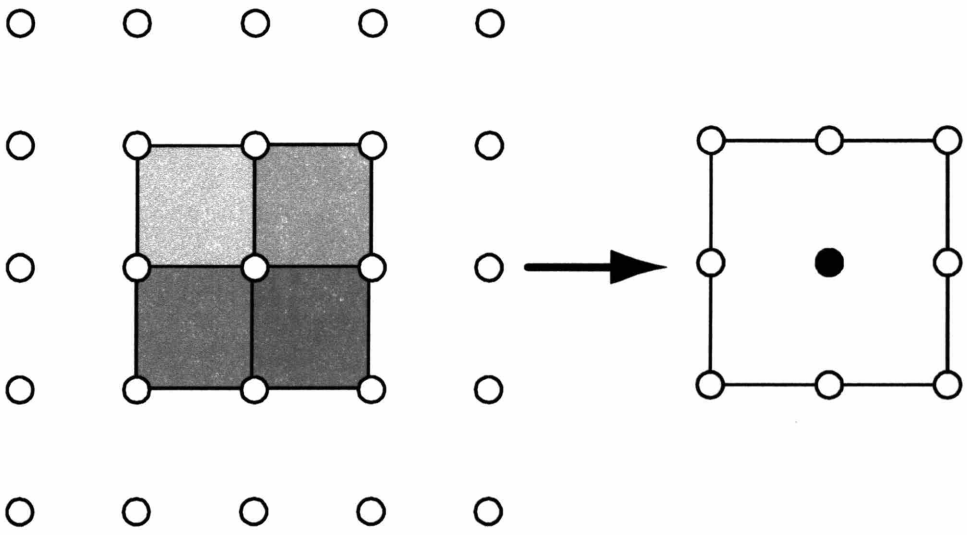


Figure 4-2: Four quadralateral elements used in construction of the local patch. The resulting patch element is an approximation of a larger quadralateral element.

It is important to note that $\nabla \mathbf{v}_{kl}$ is treated as data in eq. 4.8. By looking at the linear elements contributing to the patch, one can easily see the quadralateral form that arises from $\mathbf{P}$ (fig. 4-2). Considering the case of a patch constructed from four bilinear rectangular elements illustrated in figure 4-2, one can easily see that a biquadratic element results, described by a biquadratic basis function. Although these four-element patches are the primary choice for implementation in the viscoelastic solver, it is instructive to see the effects of increasing the number of elements in the patch. In the limit as the number of elements in the patch approaches the total number of elements in the system, the global minimization equation is recovered (eq. 4.10) as

$$\lim_{m \to net} \sum^{m} \mathbf{P}^T \mathbf{P} \mathbf{G}_{kl} - \sum^{m} \mathbf{P}^T \nabla \mathbf{v}_{kl} = \int_{\Omega_e} (\mathbf{G}_{kl} - \nabla \mathbf{v}_{kl}) \, \psi d\Omega \tag{4.10}$$

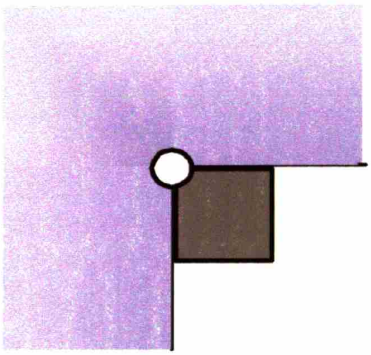
Here *net* is the total number of elements in the computational domain, Ω. Therefore, the patch algorithm is consistent with the global minimization used in the DEVSS-G formulation for the viscoelastic system of equations as well as here in the alternate

decoupled formulation described in Section 4.2.1.

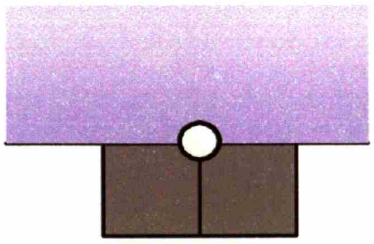
While the global minimization of $\mathbf{G}$ is not a boundary-value problem and hence does not require any information of the geometrical boundaries, since the patches used to compute the components of $\mathbf{G}$ are constructed from the surrounding elements, it is necessary to give special treatment to the nodes lying on the boundaries. When considering patches consisting of four quadralateral elements, there are three possible cases for the patch configuration of an element lying on a boundary: a corner node, a side node, and an inverted corner node. The three cases are presented in figure 4-3. Each case has four possible orientations with respect to the patched node. Notice that the three configurations have one, two, and three elements, respectively, from which the local patch is constructed. The special treatment necessary to correctly represent the information present is somewhat trivial, though necessary. When looping over the elements n in eq. 4.8, the correction for the non-existent elements is simply to set the values of $\nabla \mathbf{v}_{kl}$ to zero on all of the nodes in the element, whether or not the nodes are present. This will correctly weight the information in the local patch when not all contributing elements are present.

4.3 Comparison of Patch and Least Squares Formulations

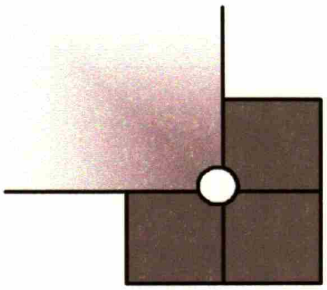
In order to understand best the effects of implementation of both the decoupled global minimization and local minimization methods, analysis in terms of accuracy and performance was carried out using the fully coupled DEVSS-G method as a benchmark. The flow in the periodic wavy walled channel was chosen as an optimal test problem for the comparisons since this flow exhibits a mixture of shear and elongational flow without the existence of any geometrical singularities. Figure 4-4 is a schematic diagram of the physical geometry modeled in the calculations. Here $H = 0.8$ is the channel height at the widest point; $H_w = 0.2$ is the swell height of the channel wall; $L_p = 1.0$ is the length of



(a) corner



(b) side



(c) inverted corner

Figure 4-3: Types of patch element configurations for a mesh of quadralateral elements. Each of the three cases has four possible orientations relative to the patched node.

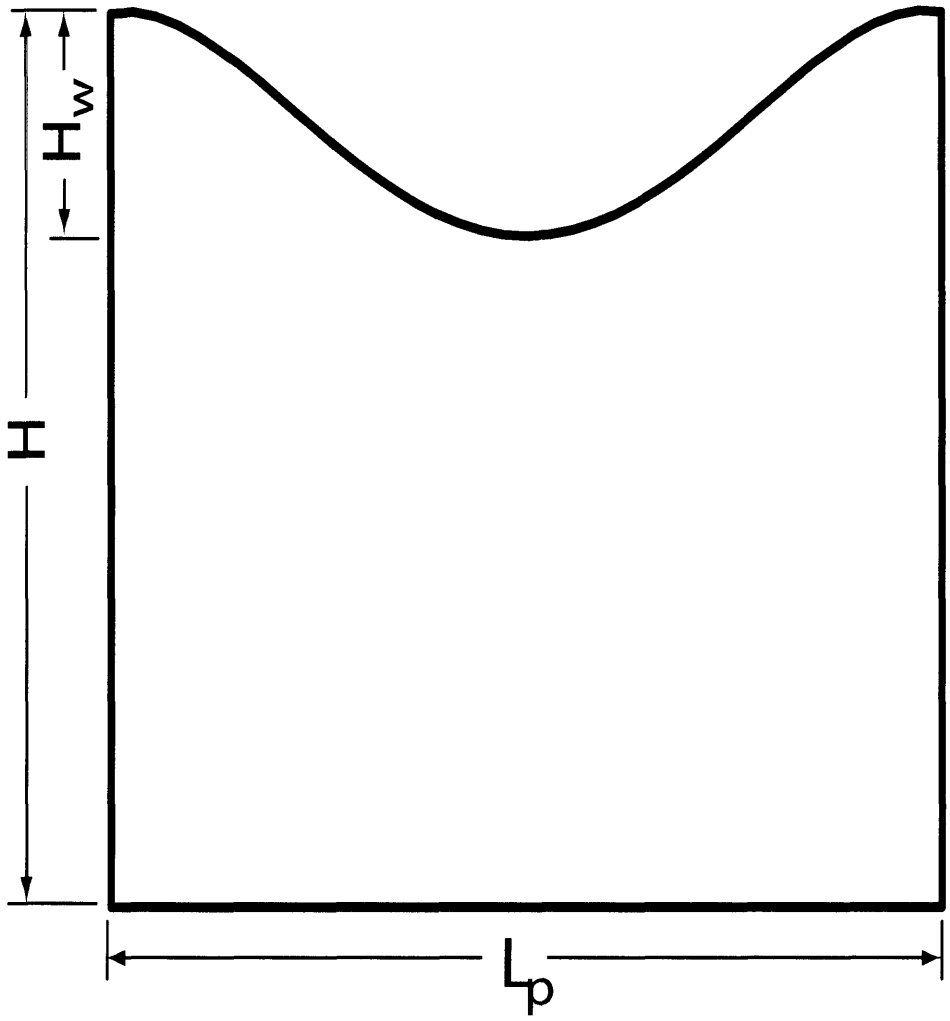
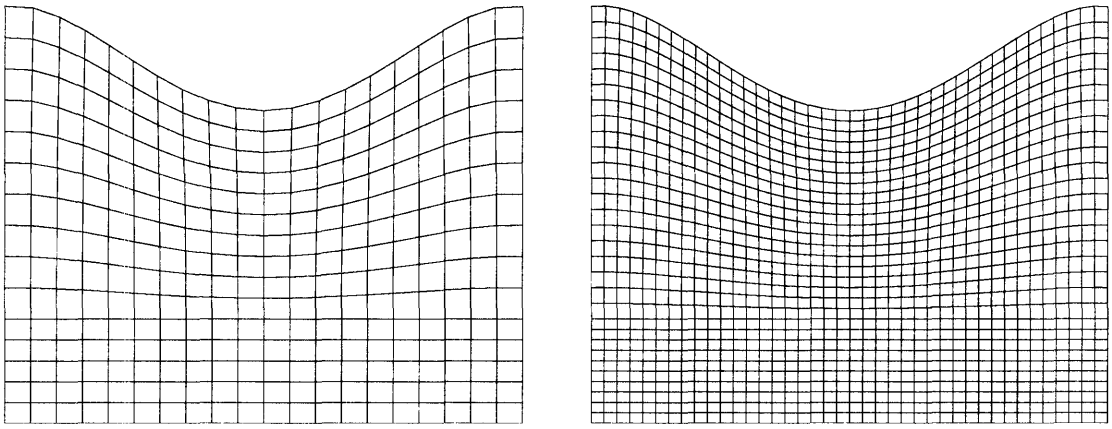


Figure 4-4: Schematic diagram of a wavy walled channel. $H = 0.8$ is the channel height at the widest point; $H_w = 0.2$ is the amplitude of the sine wave describing the undulations in the top wall; $L_p = 1.0$ is the length of a periodic section of the channel.

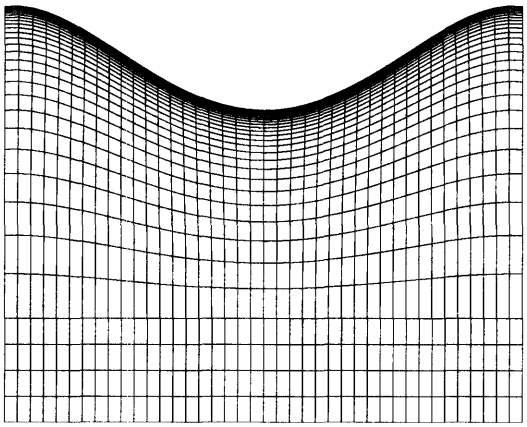
a periodic section of the channel; $2R_c = 0.3$ is the amplitude of the sine wave describing the undulations in the channel wall. Flow in figure 4-4 is from left to right. Periodic boundary conditions are applied on the inlet and outlet surface of the geometry. Zero velocity boundary conditions are enforced on the channel wall. A symmetry boundary condition is enforced on the channel centerline. Meshes used to discretize the physical geometry are shown in figure 4-5. Meshes M1, M2, and M3 contain successive levels of refinement. Mesh M2a has approximately the same number of elements as mesh M2, but with grading in the vertical direction concentrated near the channel wall where the gradients are typically the largest. Calculations were performed for an Oldroyd-B fluid with solvent to total viscosity ratio of $\beta = 0.5$ over a range of Deborah numbers, $\mathrm{De} = \lambda \langle v \rangle / h$, where λ is the relaxation time of the polymer, $\langle v \rangle$ is the average velocity in the narrowest section of the channel and $h = H - H_w$ is the height of the channel in the same region. The time constant λ is used to vary De in the simulations.

To understand better the effects of decoupling the velocity gradient interpolant equations from the momentum and mass continuity equations, comparisons of the solutions from the decoupled global minimization method and the fully coupled method are presented here. The L_2 norm of a vector $\mathbf{x}$ is defined as $\|\mathbf{x}\| \equiv \sqrt{\mathbf{x}^T\mathbf{x}}$. The comparison is carried out by monitoring the L_2 norm of the velocity field, $\|\mathbf{v}\|$, during startup of flow in the wavy-walled channel for both the decoupled global minimization method and the fully coupled method. Figure 4-6 is a plot of the difference of $\|\mathbf{v}\|$ for the two methods. The L_2 norm of the stress field, $\|\boldsymbol{\tau}\|$, is included to show when the steady state solution has been reached for the fully coupled method. As is expected from decoupling the computation of the velocity gradient interpolant equations from the flow field equations, there is some initial difference in the computed flow field. However, as the stress field evolves, the difference in $\|\mathbf{v}\|$ monotonically decreases approaching zero as the flow reaches steady state. Comparison of the solutions of the flow fields and pressure fields show that the solutions produced by the two methods do agree within numerical precision at steady state. This if of course the anticipated result, since the steady-state

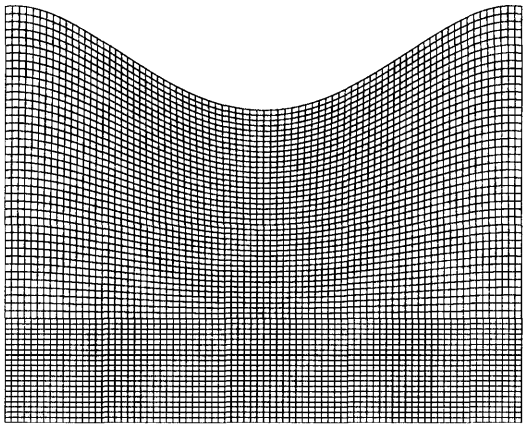


(a) M1

(b) M2



(c) M2a



(d) M3

Figure 4-5: Meshes for the wavy-walled channel geometry. M1: 300 elements; M2: 1200 elements, M2a: 1150 elements; M3: 4800 elements

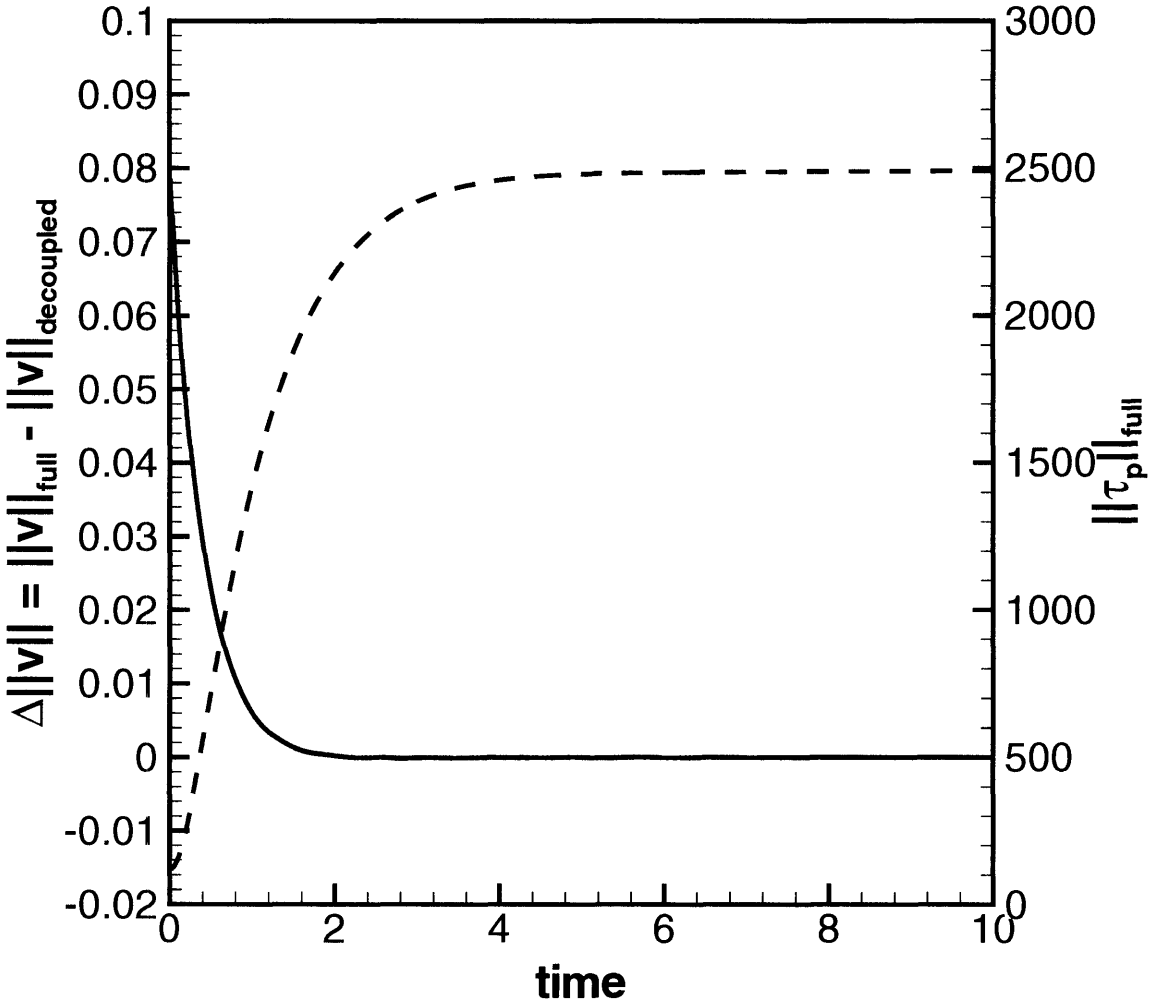


Figure 4-6: Error associated with decoupling of the global minimization equations for the velocity gradient interpolants from the momentum and mass continuity equations in startup of flow in the wavy-walled channel of an Oldroyd-B fluid of $\beta = 0.5$ and De $= 0.7$. —: $\Delta \|\mathbf{v}\|$; – –: $\|\boldsymbol{\tau}\|_{full}$.

equations for both the decoupled global minimization and the fully coupled method are identical. The conclusion can therefore be drawn that decoupling the computation of the velocity gradient interpolant from the equations describing the flow field has no effect on the computed steady-state solution.

Since a difference in the computed flow field is present prior to attaining steady state, the convergence with successive iterations of the decoupled method was also studied. For this study, the stress field was determined using the fully coupled method for the steady-state flow in the wavy-walled channel for De = 0.7. A simulation using the decoupled method was then carried out with zero initial values for the velocity, pressure, and velocity gradient interpolant fields and the steady-state stress field as computed from the fully coupled simulation at De = 0.7. In this manner, the number of iterations necessary for the decoupled global minimization method to converge to the fully coupled method solution was determined. Figure 4-7 is a plot of $\|\mathbf{v}\|$ for the decoupled method. The dashed line is the value of $\|\mathbf{v}\|$ for the fully coupled method. As is shown here, the decoupled method requires 17 iterations for the flow field to reach the steady-state solution computed by the fully coupled method.

Armed with a better understanding of the effects of decoupling the computation of the velocity gradient interpolant from the flow equations, comparison of steady-state solutions of the global and local minimization methods can now be made. For the flow of a Newtonian fluid in the wavy-walled channel, the solutions obtained with the global and the local minimization methods show little variation. As De increases, small differences in the computed flow fields and pressure fields from the two methods begin to appear. Figure 4-8 shows the contours of the two components of velocity and the pressure as computed by the two methods. To aid in identification of differences in the two flow fields, overlays of the contours of of velocity and pressure computed from the two methods are included in figure 4-9. As is apparent from the overlays of the velocity and pressure contours, only minute differences are present in the solutions computed from the two methods. The pressure field is clearly the most sensitive to the differences in the

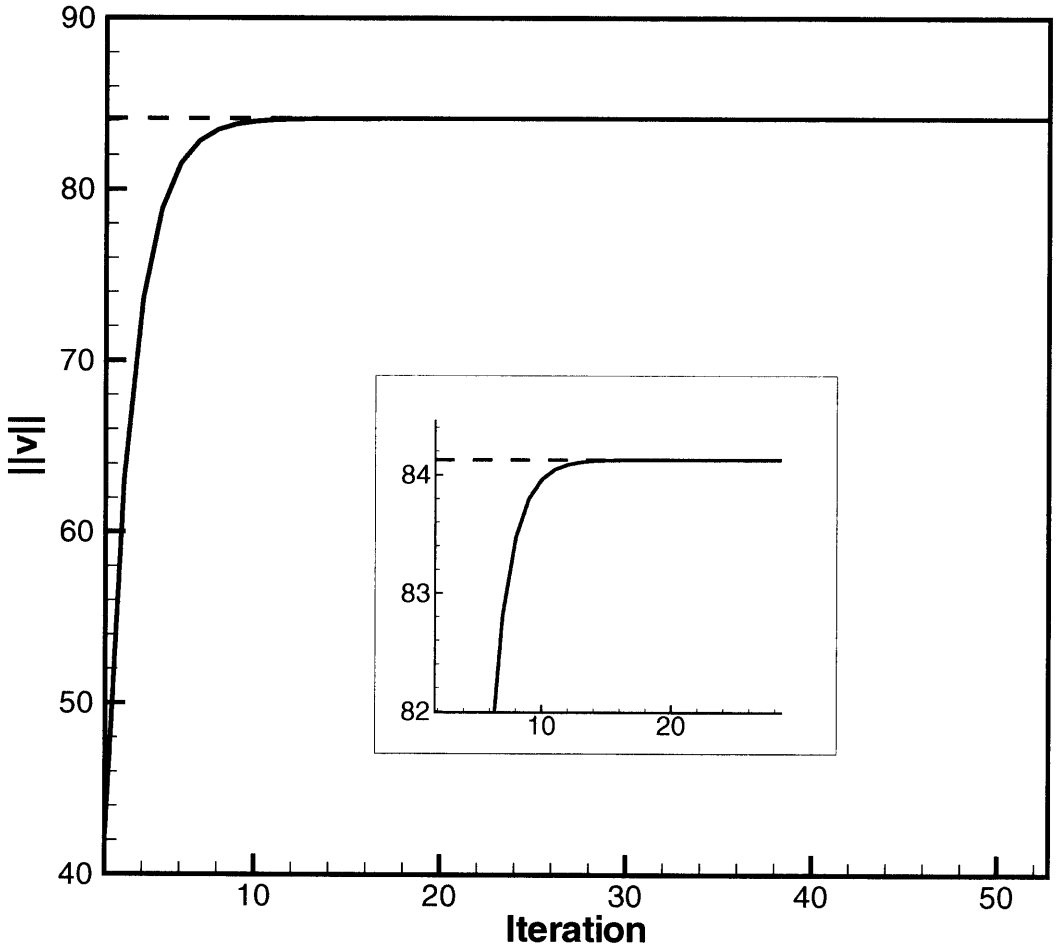
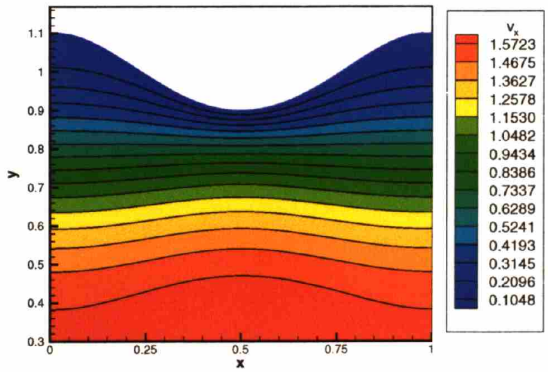
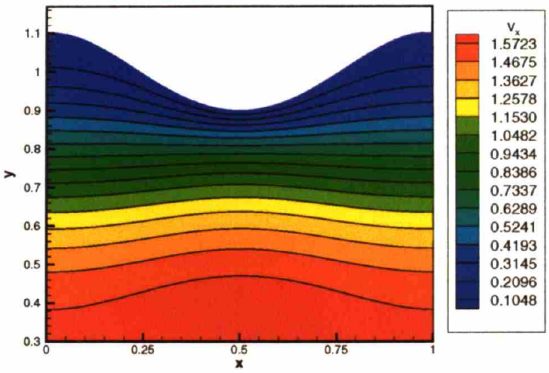


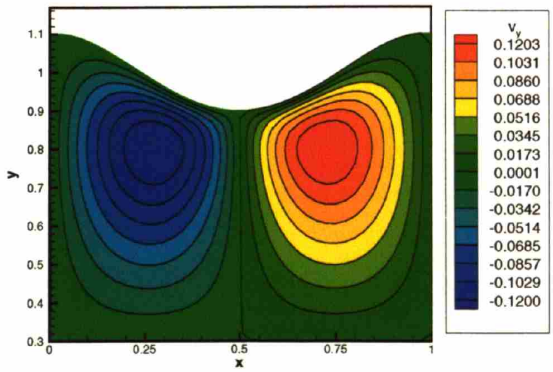
Figure 4-7: Convergence of the flow in the wavy-walled channel using the decoupled global minimization method for computation of the velocity gradient interpolant with a fixed stress field. Stress field computed using the fully coupled method for the flow of an Oldroyd-B fluid of $\beta = 0.5$ with $\mathrm{De} = 0.7$.



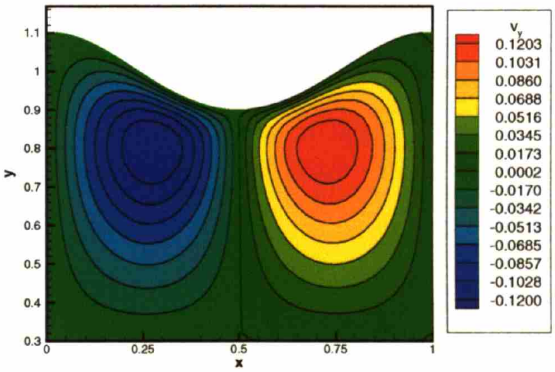
(a) global: $\mathbf{v}_x$



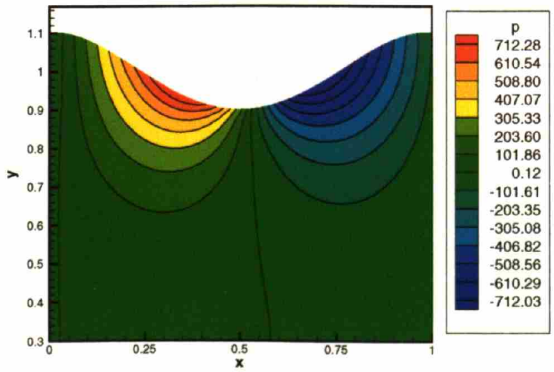
(b) local: $\mathbf{v}_x$



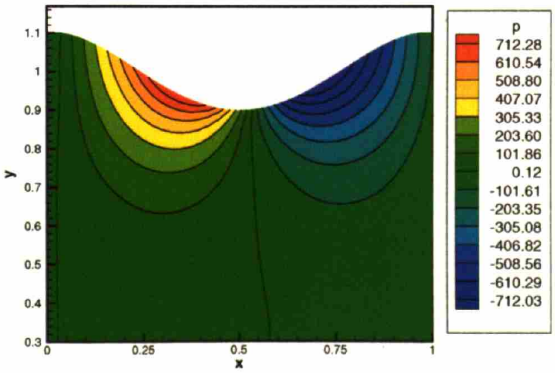
(c) global: $\mathbf{v}_y$



(d) local: $\mathbf{v}_y$

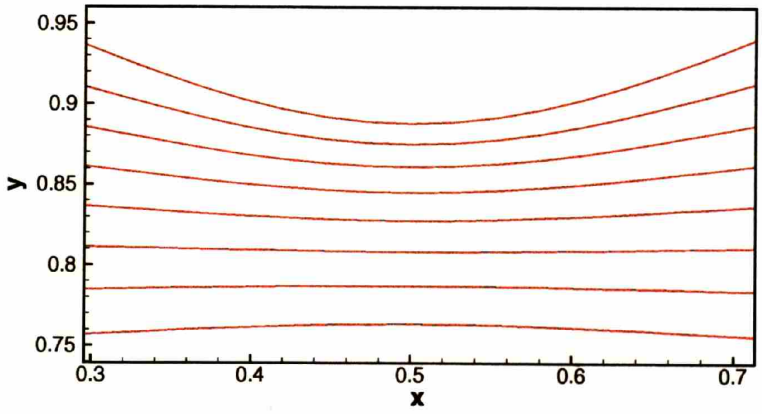


(e) global: p

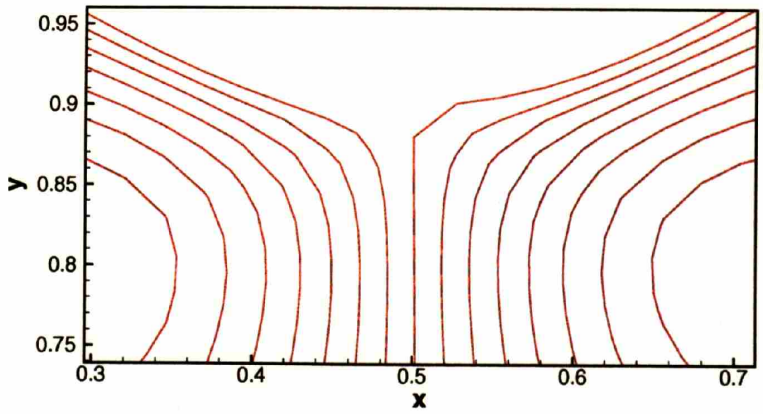


(f) local: p

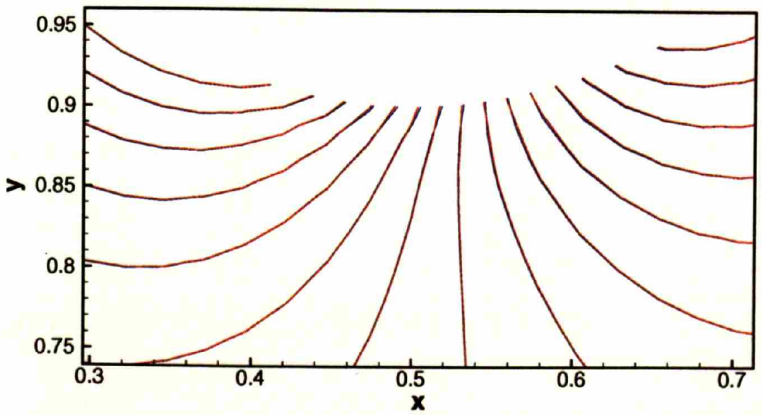
Figure 4-8: Velocity and pressure contours computed with the global and local smoothing methods with mesh M2a for $\mathbf{De} = 0.1$.



(a) $\mathbf{v}_x$



(b) $\mathbf{v}_y$

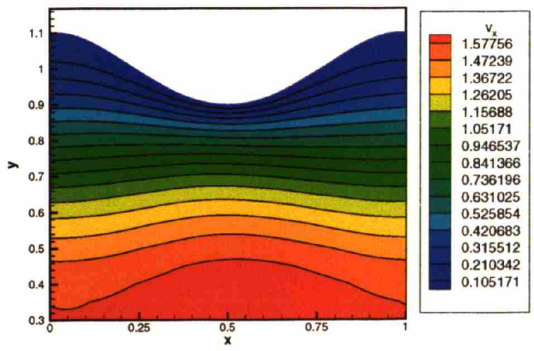


(c) p

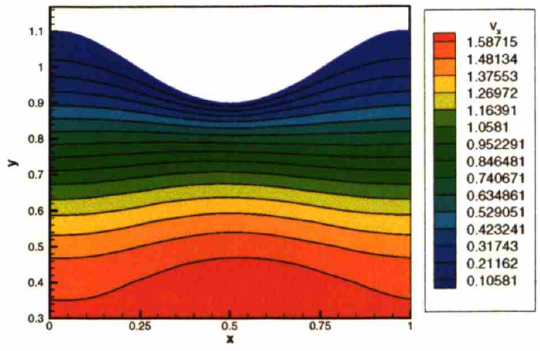
Figure 4-9: Comparison of the velocity and pressure contours computed with the global and local smoothing methods with mesh M2a for $\mathbf{De} = 0.1$. Blue: global method; red: local method.

approximations of the two methods, but still exhibits only small deviations concentrated near the wall of the narrowest region of the channel. As the viscoelasticity of the flow increases, the differences in the flow field and pressure field computed with the two methods become more pronounced. Figure 4-10 shows the velocity and pressure contours as computed with the two methods for De = 1.0. Overlays of the contour fields for velocity and pressure for the De = 1.0 case are included in figure 4-11. In contrast to the De = 0.1 case, the overlay of the pressure contours for the De = 1.0 case shows much more significant variation in the pressure fields computed with the two methods. The variations are still concentrated near the wall of the narrowest section of the channel but now remain pronounced much further away from the wall. In addition, the velocity fields computed with the local smoothing method are now beginning to show significant deviation from the global method.

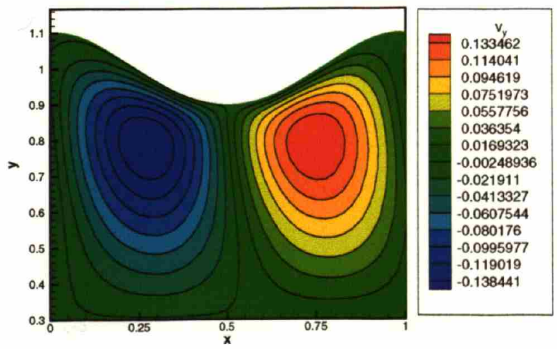
Clearly the global and local minimization methods produce different results for the flow field in the wavy-walled channel as viscoelasticity increases. As with the computed flow fields and pressure fields, comparison of the differences in the computed values of the components of $\mathbf{G}$ show increasing differences as the viscoelasticity is increased. Contour plots of the components of $\mathbf{G}$ for the global and local minimization cases for De = 0.1 are shown in figure 4-12. Figure 4-13 shows overlays of the contours of the components of the velocity gradient interpolant computed using the two methods in the region near the wall of the narrow section of the channel. These comparisons show that the most significant variation between computed values of the global and local smoothing methods occurs near the wall of the channel in the narrowest section. While the contour plots of the components of $\mathbf{G}$ are still quite similar at De = 0.1, the overlays of the solutions show small differences beginning to arise between the computed values of the two methods near the wall of the narrowest region of the geometry, the region where the gradients are the largest in magnitude. As with the computed values of velocity and pressure, the approximations for $\mathbf{G}$ of the two methods show an increase in variation as the viscoelasticity is increased. Contour plots of the components of $\mathbf{G}$ for the global



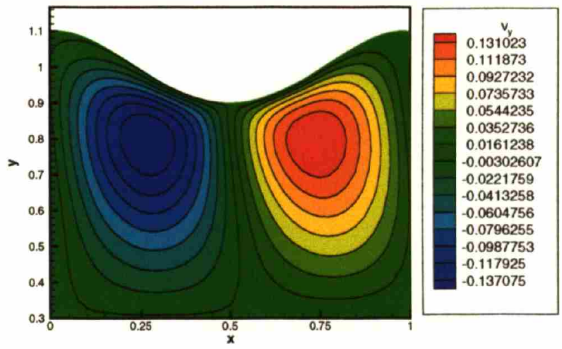
(a) global: $\mathbf{v}_x$



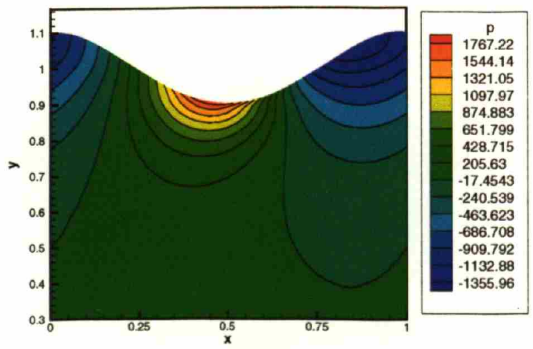
(b) local: $\mathbf{v}_x$



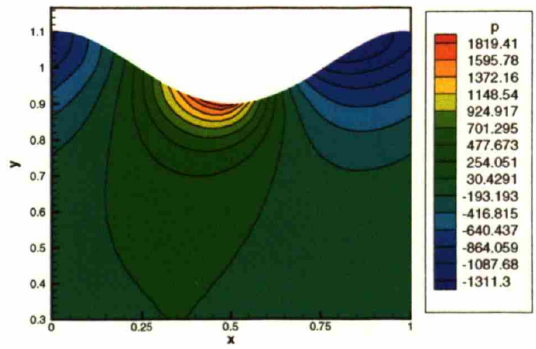
(c) global: $\mathbf{v}_y$



(d) local: $\mathbf{v}_y$

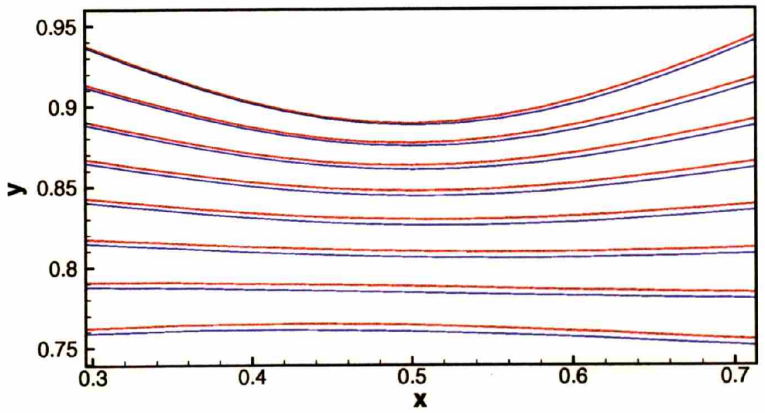


(e) global: p

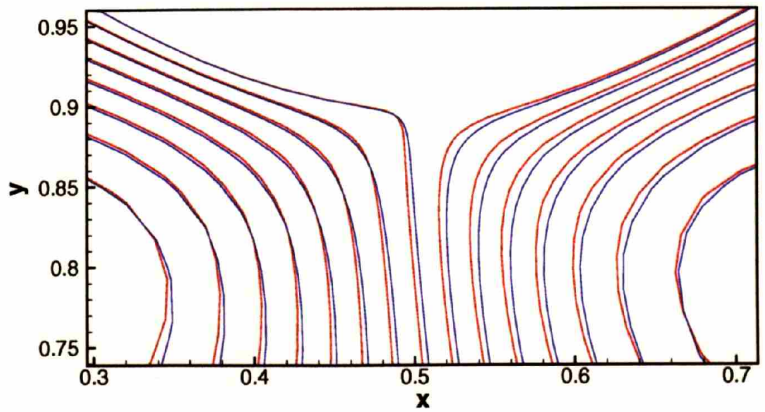


(f) local: p

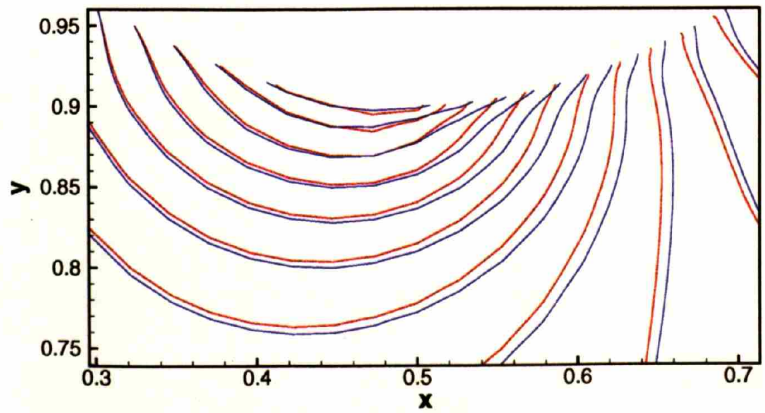
Figure 4-10: Velocity and pressure contours computed with the global and local smoothing methods with mesh M2a for $\mathbf{De} = 1.0$.



(a) $\mathbf{v}_x$

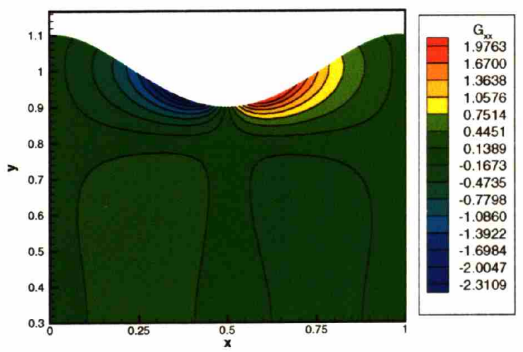


(b) $\mathbf{v}_y$

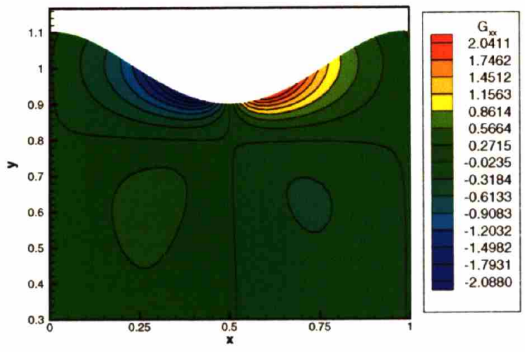


(c) p

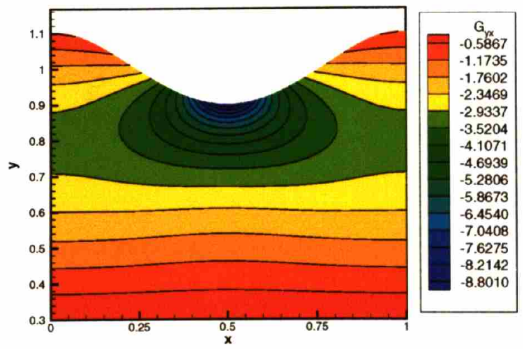
Figure 4-11: Comparison of the velocity and pressure contours computed with the global and local smoothing methods with mesh M2a for $\mathbf{De} = 1$. Blue: global method; red: local method.



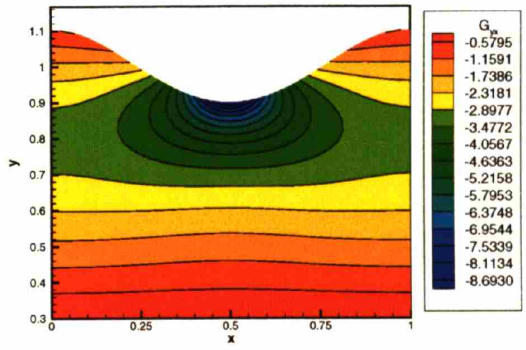
(a) global: $\mathbf{G}_{xx}$



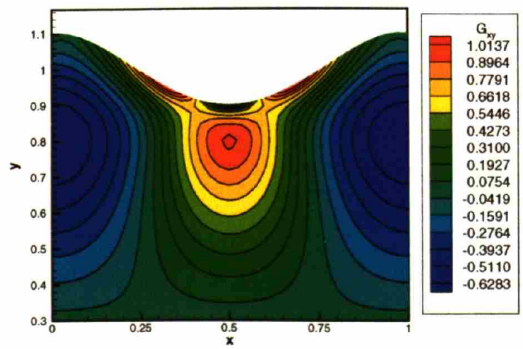
(b) local: $\mathbf{G}_{xx}$



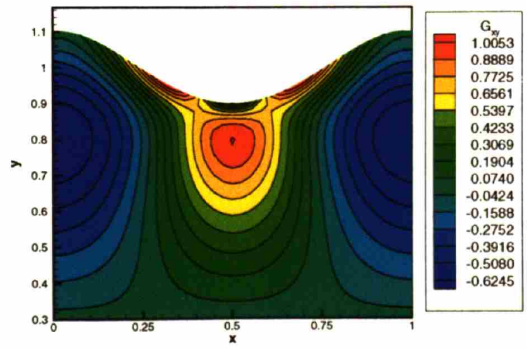
(c) global: $\mathbf{G}_{yx}$



(d) local: $\mathbf{G}_{yx}$

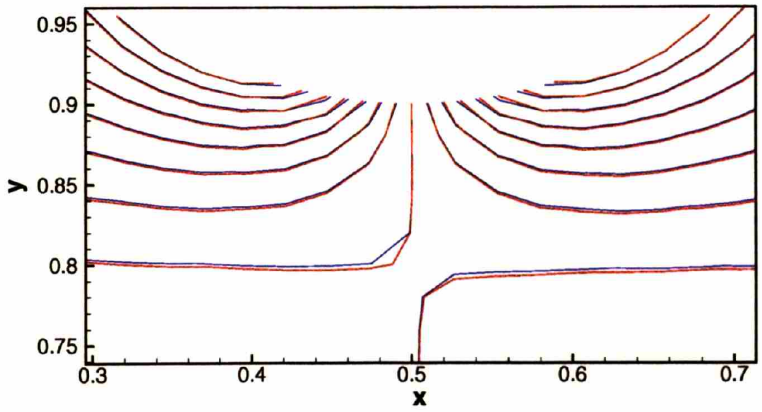


(e) global: (G_{xy}

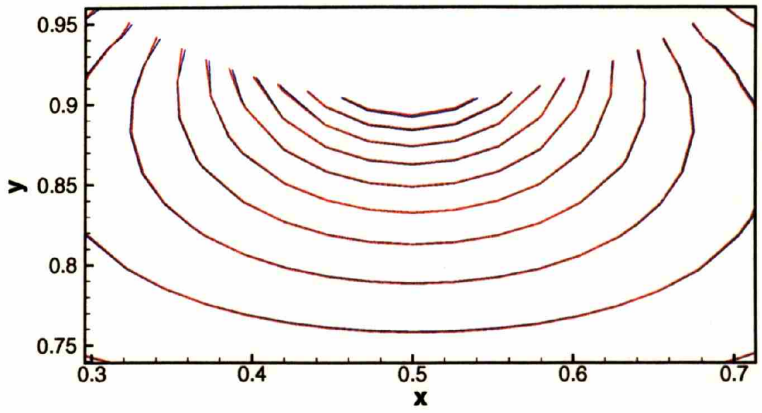


(f) local: $\mathbf{G}_{xy}$

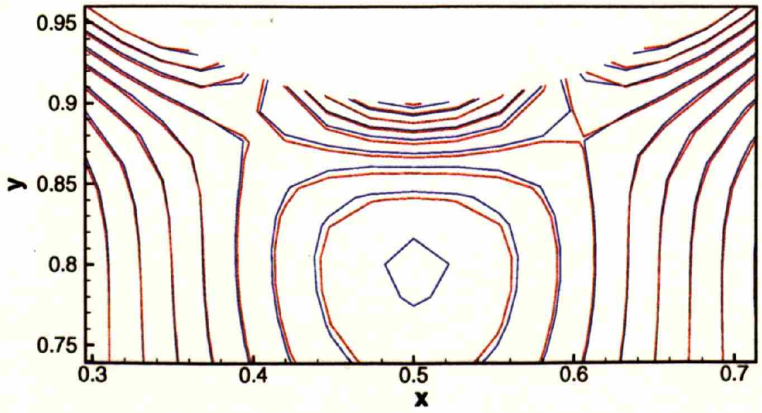
Figure 4-12: Contour plots of the components of G computed using the global and local minimization methods with mesh M2a for De=0.1.



(a) $\mathbf{G}_{xx}$



(b) $\mathbf{G}_{yx}$



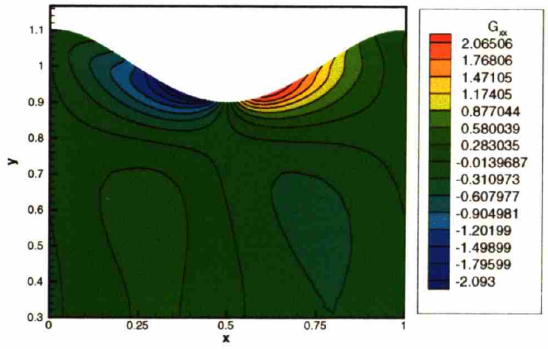
(c) $\mathbf{G}_{xy}$

Figure 4-13: Comparison of the contours of the components of $\mathbf{G}$ computed with the global and local smoothing methods with mesh M2a for $\mathbf{De} = 0.1$. Blue: global method; red: local method.

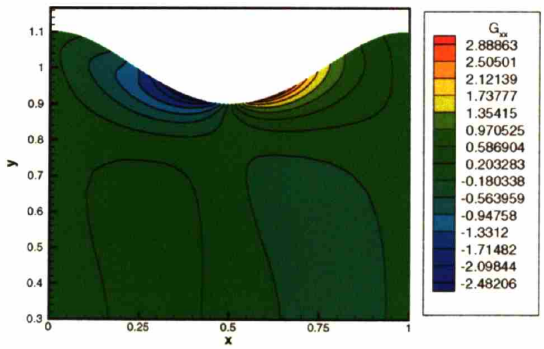
and local minimization cases for De = 1.0 are shown in figure 4-14. Overlays of the contour fields for the components of $\mathbf{G}$ for the De = 1.0 case are included in figure 4-15. From the contour overlay plots it is apparent that the increased variation with increasing viscoelasticity is still concentrated around the wall of the narrowest region of the channel; this is the region of the largest viscoelastic stresses. While variation in $\mathbf{G}_{xx}$ is not significant at the wall in the narrow section, the local method predicts much higher values at the wall just upstream and downstream of the narrow region. Comparisons of $\mathbf{G}_{yx}$ and $\mathbf{G}_{xy}$ also show dramatic differences between solutions of the global and local approximations. In figure 4-15(b) the local smoothing method is shown to predict a much higher gradient in $\mathbf{G}_{yx}$ near the wall of the narrowest region of the channel. In figure 4-15(c) the local smoothing method is shown to predict higher values of $\mathbf{G}_{xy}$, most apparent just downstream of the narrowest region of the channel.

As the viscoelasticity is further increased, the local smoothing method shows even greater differences in the computed flow field as well as the velocity gradient interpolant fields. With mesh M2a, the local smoothing method fails to reach steady state above De = 1.0, whereas the global smoothing method continues to reach steady state up to De = 1.5. Figure 4-16 shows that whereas the local smoothing method reaches steady state in approximately the same number of iterations as the global smoothing method, a discrepancy in the flow field exists. The increased discrepancy of the flow field over a range of De calculated from the local smoothing method is qualified by using the fixed stress field analysis described previously. Figure 4-17 is a plot of the difference in the L_2 norms of the flow field with increasing viscoelasticity as predicted by the global and local smoothing methods using a fixed stress field. Note that with the fixed stress field calculated from the fully coupled method, temporal convergence of the local smoothing method can be obtained for higher De using the M2a mesh. Although the error in the patch method approaches a limiting value as De$\rightarrow$ 0, the error grows rapidly as De is increased.

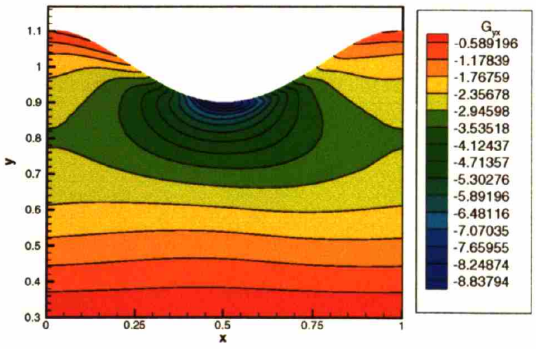
To understand better if the local smoothing method is convergent with increased



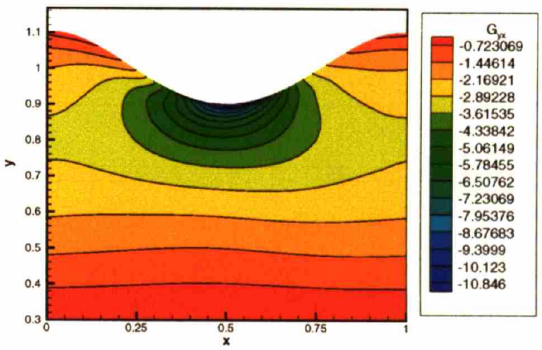
(a) global: $\mathbf{G}_{xx}$



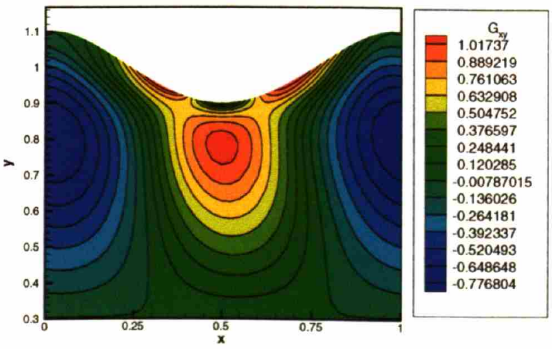
(b) local: $\mathbf{G}_{xx}$



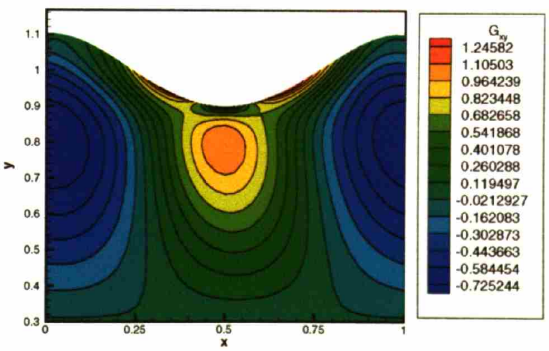
(c) global: $\mathbf{G}_{yx}$



(d) local: $\mathbf{G}_{yx}$

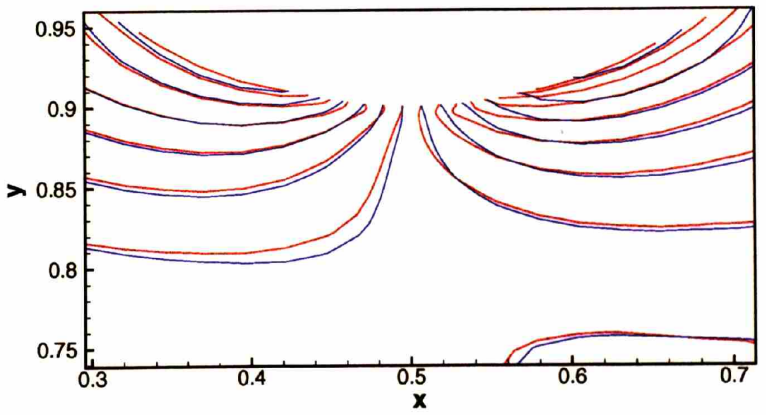


(e) global: $\mathbf{G}_{xy}$

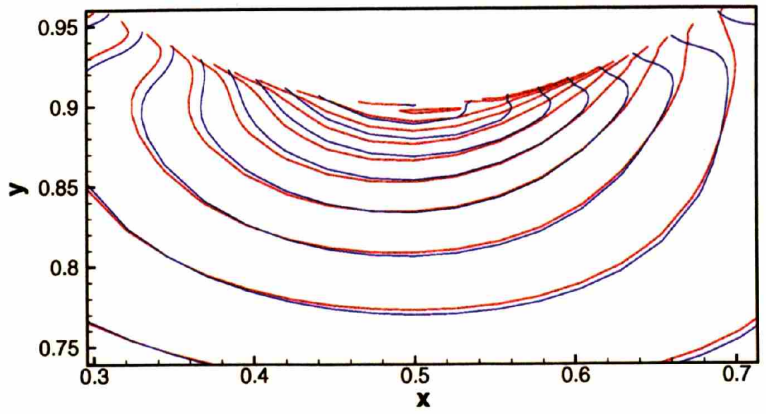


(f) local: $\mathbf{G}_{xy}$

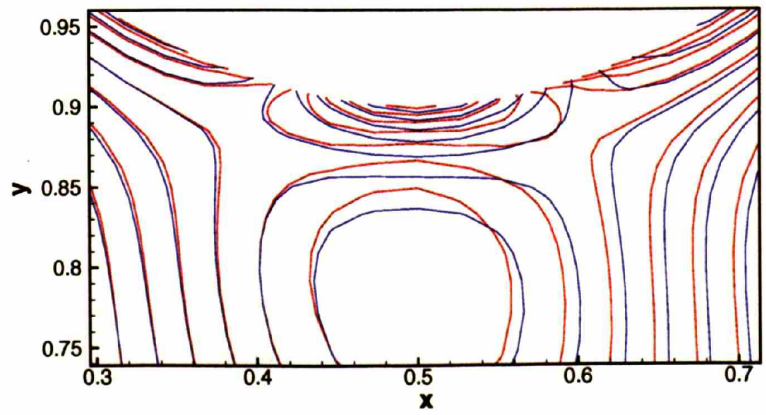
Figure 4-14: Contour plots of the components of G computed using the global and local minimization methods with mesh M2a for De=1.0.



(a) $\mathbf{G}_{xx}$



(b) $\mathbf{G}_{yx}$



(c) $\mathbf{G}_{xy}$

Figure 4-15: Comparison of the contours of the components of $\mathbf{G}$ computed with the global and local smoothing methods with mesh M2a for $\mathbf{De} = 1.0$. Blue: global method; red: local method.

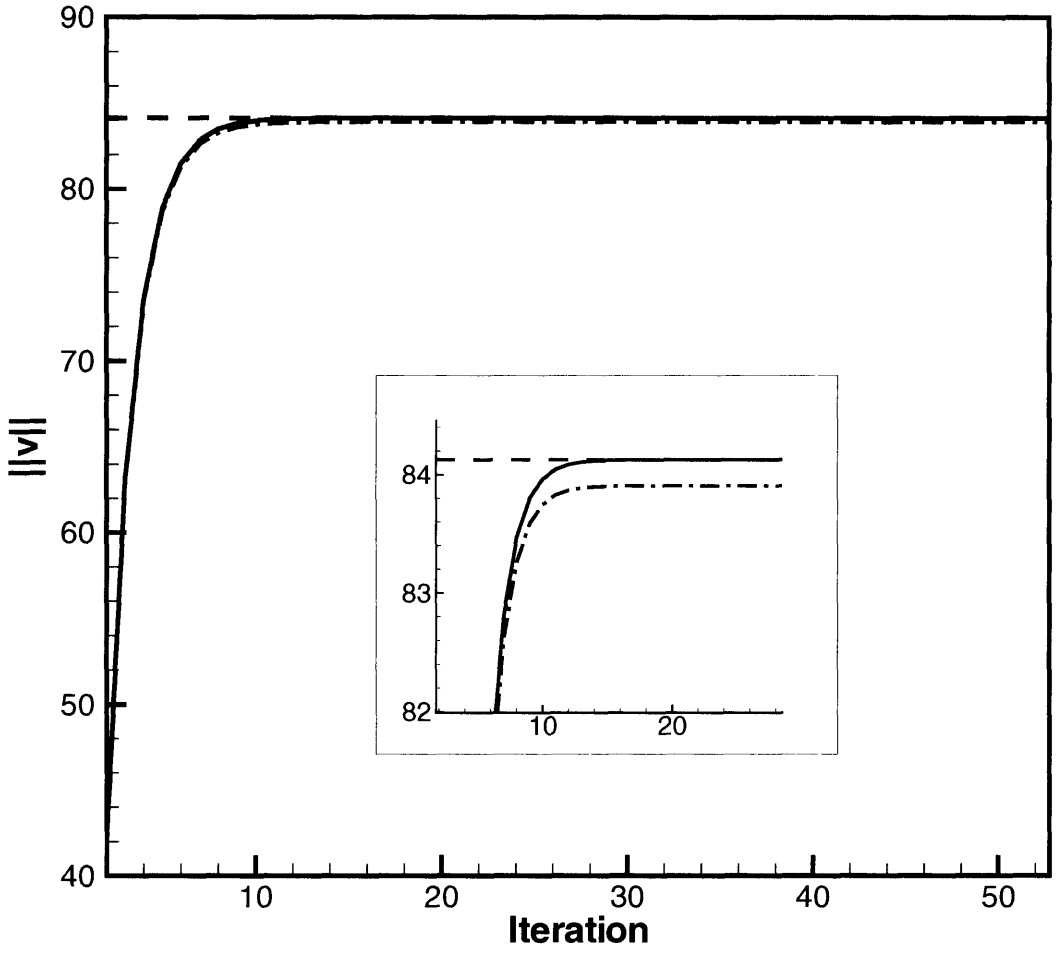


Figure 4-16: Convergence of the L_2 norm of the velocity field in the wavy-walled channel using the decoupled local smoothing method ($- \cdot -$) for computation of the velocity gradient interpolant with a fixed stress field. Stress field computed using the fully coupled method for the flow of an Oldroyd-B fluid of $\beta = 0.5$ with De $= 0.7$. Decoupled global minimization method ($-$) is included for comparison.

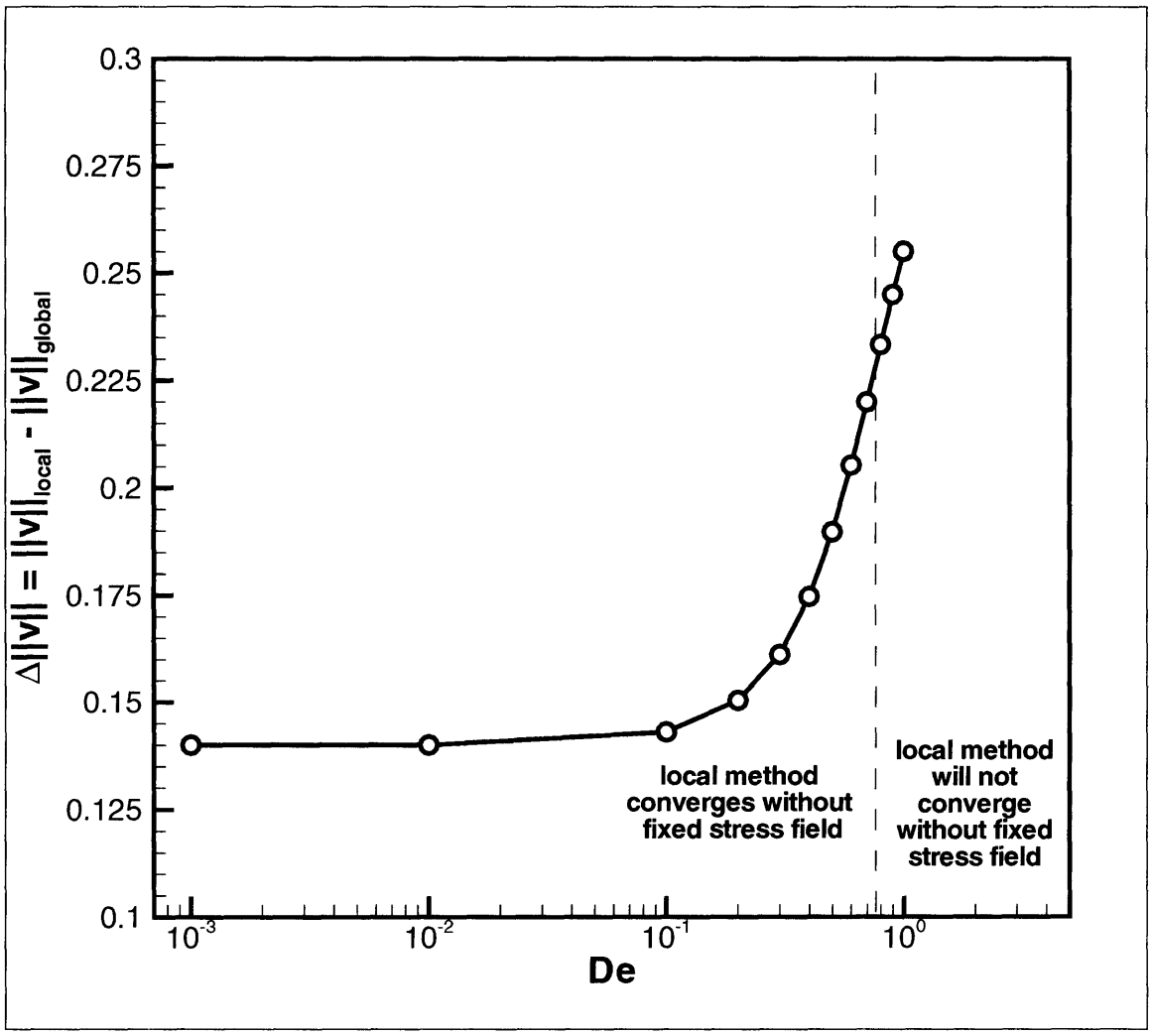


Figure 4-17: Difference in L_2 norms of the velocity computed from the global and local smoothing methods using a fixed stress field over a range of De for an Oldroyd-B fluid with $\beta = 0.5$. Mesh M2a is used in the calculations.

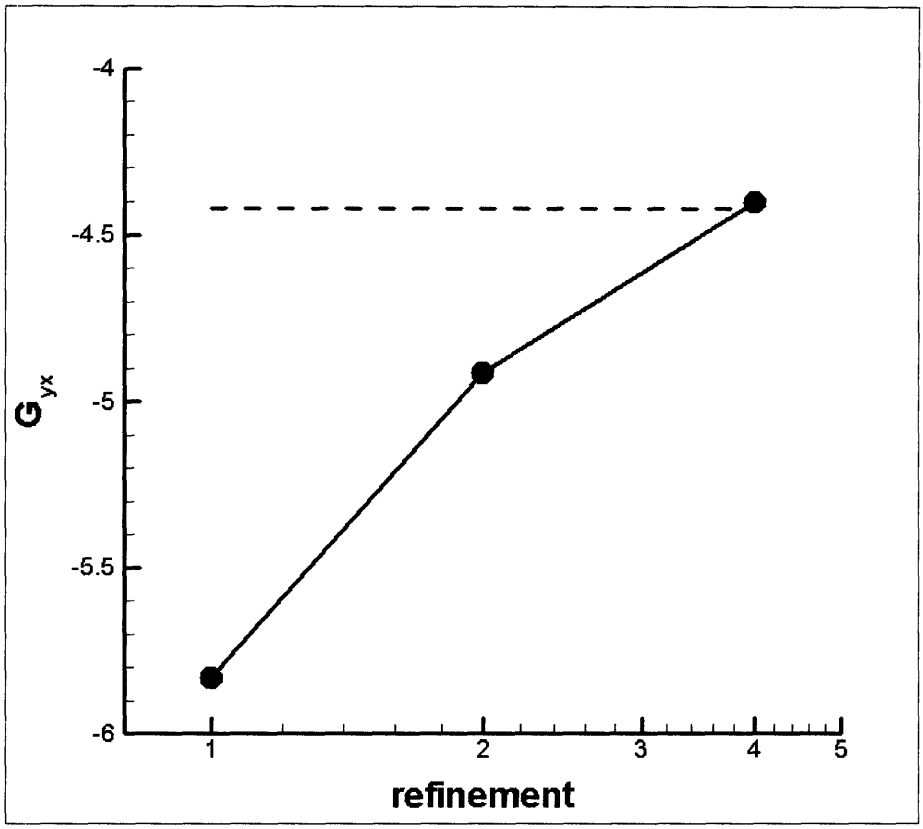


Figure 4-18: Mesh convergence study of the local smoothing method. Refinement r is inversely proportional to the element dimensions in the mesh. (– –): global method with mesh M1; (○): local method. $\mathbf{G}_{yx}$ on channel wall at $x = 0.65$.

mesh resolution, calculations were performed with meshes M1, M2, and M3 using the local smoothing method. Comparison of the solutions of the global smoothing method for the three meshes showed virtually no difference; hence, the global smoothing method is considered to be converged with the M1 mesh. Comparison was made between the solutions of the global smoothing method with mesh M1 and the local smoothing method with meshes M1, M2, and M3. Figure 4-18 shows the value of $\mathbf{G}_{yx}$ at the wall just downstream of the narrowest part of the channel for the two methods. Refinement r is inversely proportional to the dimensions of the elements in the mesh such that doubling r results in doubling the number of elements in both the x and y directions, effectively

quadrupling the total number of elements. It is shown here that the local smoothing method converges to the global method with increasing mesh resolution. This super linear convergence agrees with the theory presented in [90]. Using the local smoothing method it is possible to obtain steady-state solutions at higher De with increased mesh resolution. In order determine the amount of refinement needed to converge to the solution of the global smoothing method when utilizing the local smoothing method, solutions found for the three meshes, M1, M2, and M3, were compared to the solution from the global smoothing method, since the level of refinement necessary to converge to the global smoothing method's solution was not necessarily required in all regions of the flow domain. Figure 4-19 identifies where and approximately how much additional refinement is necessary for the local smoothing method to converge to the global smoothing method for the flow in the wavy-walled channel with $\mathrm{De} = 1.0$. The resulting mesh has approximately seven times the number of unknowns of the equivalent mesh for the global smoothing method.

By performing an operation count for the local smoothing method and global smoothing method for the equivalent meshes needed to reach similarly converged solutions with both methods, better understanding of the relative computational costs of the two methods can be gained. Take N to be the total number of unknowns in the system. The operation count for both methods can be broken up into the count for the subproblems of the calculation of the velocity gradient interpolant and the calculation of the velocity and pressure. Consideration of the operation count of the latter subproblem is necessary due to the effect of the increase in mesh resolution needed in the local smoothing method on this calculation. For operation count estimation, it is assumed that a Krylov subspace method with suitable preconditioner can be found to solve the large linear equation sets in the velocity-pressure subproblem [16] and the global least squares minimization of $\mathbf{G}$ subproblem. The constant, b, in the operation count of the Krylov solver is assumed to scale with the total number of unknowns in the subproblem. For the patch algorithm, it is assumed that the evaluation of each individual patch requires approximately 10 op-

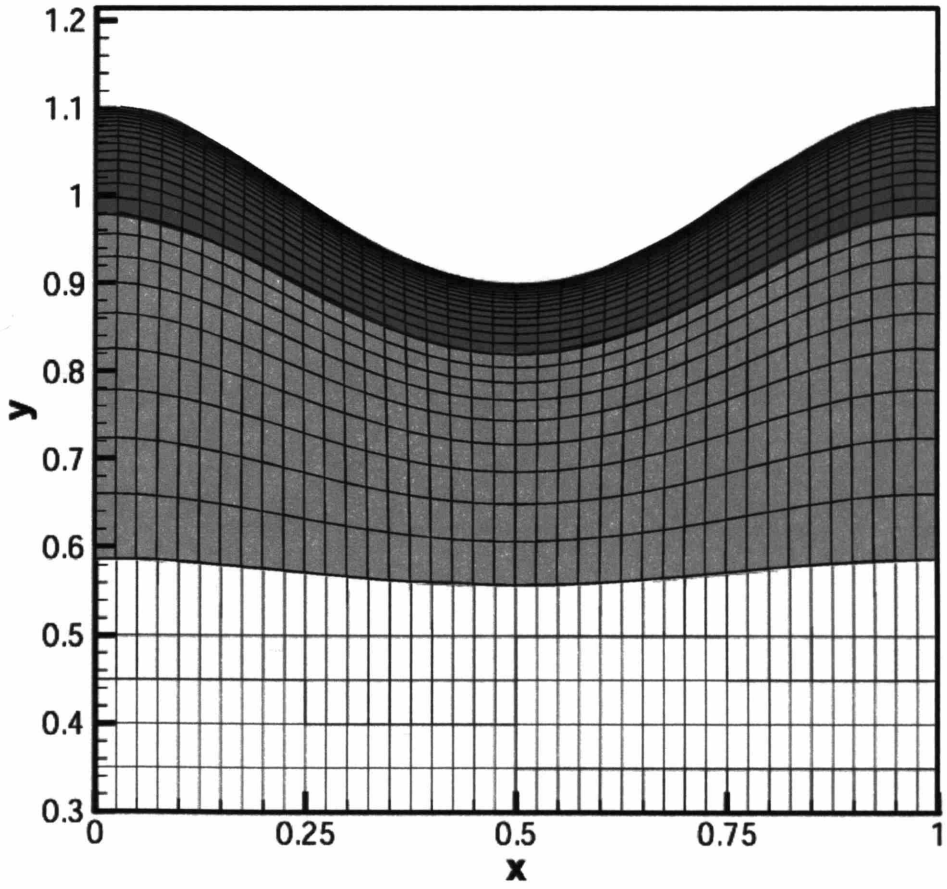


Figure 4-19: Refinement necessary when using local smoothing method to converge to global smoothing method solution. Shaded regions highlight areas of refinement. The darker shaded region requires 4 times refinement in both the x and y directions. The lighter shaded region requires 4 times refinement in x and 2 times in y. The unshaded region requires no refinement.

erations. The components of **G** make up approximately one third of the total number of unknowns in the flow problem of the mixed finite element formulation. This is true for both the two-dimensional and three-dimensional problems. In the two-dimensional case, for the global smoothing method, the subproblem for the global least squares minimization of **G** requires

$$3 * O\left((1/3 * N * 1/3) * (1/3 * b * 1/3)\right) = 0.04 * O\left(Nb\right)$$

operations. Note here that each component of **G** is solved for separately, since they have no dependence on one another. Savings in three dimensions is further enhanced by a factor of $1/3 * 3 * 3 * 1/8 * 1/8 * 8 = 3/8$. The velocity-pressure subproblem requires

$$O\left((2/3 * N) * (2/3 * b)\right) = 0.44 * O\left(Nb\right)$$

operations. The overall operation count for the global smoothing method is then

$$(0.04 + 0.44) * O\left(Nb\right) = 0.48 * O\left(Nb\right)$$

operations.

For the local smoothing method, the subproblem for the patch algorithm calculation of **G** requires

$$O\left(10 * (7 * 1/3 * N)\right) = 23 * O\left(N\right)$$

operations. The velocity-pressure subproblem requires

$$O\left((7 * 2/3 * N) * (7 * 2/3 * b)\right) = 22 * O\left(Nb\right)$$

operations. The overall operation count for the local smoothing method is then

$$(23 + 22 * b) * O\left(N\right) = 22 * O\left(Nb\right)$$

operations. Table 4.1 summarizes the operation counts for the decoupled global smoothing, decoupled local smoothing, and the fully coupled methods. While the calculation of

method	**operation count factor, x $*$ O (Nb)**
fully coupled	1
global smoothing	0.44
local smoothing	22

Table 4.1: Relative operation count for the fully coupled, decoupled global smoothing and decoupled local smoothing methods.

G using the patch algorithm is of little consequence in the overall operation count, the effect of the increased resolution required to approximate the global smoothing method on the velocity-pressure subproblem makes the overall calculation using the patch algorithm quite expensive when compared to the method utilizing global least squares minimization for **G**. Considering the operation count for the fully coupled method to be $O(Nb)$, the decoupled global smoothing method offers significant savings at roughly half of the number of operations required to reach solution.

4.4 Conclusions

In this chapter, two different methods for computing the velocity gradient from a previously computed flow field in the viscoelastic flow problem are presented in an attempt to reduce the computational load of the computation of the viscoelastic flow calculation. Both methods are based upon the DEVSS-G formulation first used by Liu et al. [54], but decouple the calculation of the velocity gradient from the flow problem. The decoupled global smoothing method is a direct extension of the DEVSS-G method, in that they both utilize a global least squares approximation for the calculation of the velocity gradient. The decoupled local smoothing method, on the other hand, utilizes a patch algorithm developed by Zienkiewicz and Zhu [90] for use in solid mechanics for determining the stress posteriori.

Since both methods utilize a decoupled form of the equation set governing viscoelastic

flow, a study of the effects of decoupling the global least squares minimization equation for $\mathbf{G}$ is performed. The most significant result of this study was that no discernible difference in the steady-state solution of the decoupled global smoothing method was found when compared to the fully coupled steady state solution. The decoupled method did however require additional iterations to reach steady state when initialized with a fixed stress field at moderate De; however, utilizing a time-dependent method automatically addresses this issue as the solution evolves in time.

Comparison of the performance of the global and local smoothing methods was studied, concentrating on differences in steady state solutions over a range of De. While only minor differences in the flow fields are present at low De, as De is increased, more dramatic differences become apparent. Eventually, above De = 1.0, the local smoothing method fails to converge to a steady-state solution, whereas the global smoothing method still is able to reach steady state up to De = 1.5 when utilizing the same finite element mesh for calculation. Calculation utilizing a fixed stress field computed with the fully coupled method shows that the error in the flow field computed with the local smoothing method grows rapidly with De. Increased mesh resolution is shown to be a possible solution to the poor performance of the local smoothing method. However, analysis of the computational cost of the increased resolution necessary to capture the same accuracy as the global smoothing method shows the local smoothing method to be quite expensive relative to the decoupled global smoothing method. On the other hand, the global smoothing method reduces the overall computational cost of the flow problem by a factor of two, clearly making it the more desirable method.

Chapter 5

Time-dependent Free-surface Formulation for Two-dimensional Viscoelastic Flows

Many technologically important problems involve free surfaces and viscoelastic fluids [49], [56], [61], [83]. One of the commonly studied free surface problems involving viscoelastic fluids is the extrusion of a polymer from a die orifice. Although this problem involves a confined die geometry in which rich dynamic behavior occurs [7], [52], [63], much interest has been centered on the dynamics that occur at and beyond the die exit [50], [53], [58], [59], [70], [78]. In particular, this problem is of great interest to the fibers and films industry. Accurate modeling of the free-surface problem would be extremely helpful in not only understanding the behavior of currently utilized process technology and polymers but also in exploring new polymers and vastly different processing technology designed through relatively inexpensive modeling performed on computers reducing the amount of costly experimentation in lab-scale and pilot plant facilities.

Implementation of free-surface boundaries through the use of deformable regions introduces significant complexity to the viscoelastic flow problem. While the use of deformable regions with steady-state solvers has been proven successful [70], [78], it introduces signif-

icant computational cost to the already expensive viscoelastic flow problem. While less computationally intensive, implementation into the decoupled time-dependent schemes may be much more troublesome than the steady-state analog. Algebraic mesh generation involving a simple interpolation scheme, [74], is one such method used in conjunction with the finite element method. However, the scheme has been shown to behave poorly, resulting in mesh distortions and crossing of elements in some instances, [20]. A successful method for updating nodal positions in two-dimensional domains is the use of a set of elliptic mapping equations developed by Christodoulou and Scriven, [21]. However, extension of the two-dimensional mapping equations to three dimensions is not straight forward.

Rasmussen and Hassager, [40], take a somewhat different approach in which the elements of the discretized domain are allowed to deform affinely given the governing equations and boundary conditions. The authors employ a remeshing technique to avoid any distortion of elements during the simulation. However, the modeling technique uses a Lagrangian frame of reference, where the simulations of interest here are problems described with an Eulerian frame of reference. Tailoring this technique to an Eulerian frame of reference yields another formulation known as the Arbitrary Lagrangian-Eulerian method in which the moving variables are segregated from the non-moving variables, [79]. Other authors have demonstrated the use of elements that lie along the computed velocity streamlines for three-dimensional solvers, [25], [58]. However, this adds yet another subproblem since the velocity field is needed to compute locations of the elements. How this substep would be implemented in a time-dependent viscoelastic solver is not clear and may cause numerical instabilities to arise. To make matters worse, there is a known temporal instability, the Hadamard instability, arising from the constitutive equation that may affect the implementation of a free-surface boundary, [45], [51]. Joseph [45] provides an extended discussion on hyperbolic equations and change of type leading to the Hadamard instability.

This chapter details a two-dimensional implementation of deformable mesh regions

with free-surface boundary conditions within the time-dependent DEVSS-G formulation discussed in Section 3.1. The method capitalizes on the decoupled, time-dependent formulation of the governing equations by including the evolution equation describing the height of the free surface in a self-contained subproblem, limiting the impact on the overall problem size from creation of the new unknowns for nodal position in the deformable region of the domain. Reduction of this impact, while not crutial in two-dimensional problems, is critical to the application of the free-surface equations in the much larger three-dimensional problems. Furthermore, application of the new physical equation in the manner presented here can be extended to include evolution equations describing other physical aspects of the system, such as the energy equation, with minimal impact to the overall problem size. Derivations of the equations describing the free-surface boundary conditions and equations used for updating node locations of deformable elements are included in Section 5.1. Comparison of solutions from the time-dependent method to those of a known steady-state method [72] are detailed in Section 5.2. Conclusions are presented in Section 5.3.

5.1 Problem Description

5.1.1 Governing Equations

The DEVSS-G method is employed here to describe the flow of a viscoelastic fluid in an unconfined geometry, given by eqs. 3.7, 3.8, 3.9, and 3.10. Development of the method is described in full in Section 3.1. The polymer extra stress is modeled using the Giesekus equation, eq. 2.26.

5.1.2 Free-surface Boundary Conditions

Boundary conditions for both the total stress on the meniscus boundary and the motion of the meniscus boundary are necessary to describe the system fully. The stress balance

on the surface consists of the tangent component and the normal component, namely

$$\mathbf{t} \cdot \boldsymbol{\pi} \cdot \mathbf{n} = 0 \tag{5.1}$$

$$\mathbf{n} \cdot \boldsymbol{\pi} \cdot \mathbf{n} = \frac{2H}{\mathrm{Ca}} \tag{5.2}$$

where $\mathbf{t}$ is the tangent to the meniscus, $\mathbf{n}$ is the outward normal to the meniscus, $\boldsymbol{\pi} = \boldsymbol{\tau} + p\boldsymbol{\delta}$ is the total stress, H is the surface curvature, $\mathrm{Ca} \equiv \eta \langle v \rangle / \sigma$ is the capillary number, and σ is the surface tension.

The kinematic boundary conditions for the meniscus boundary are written as

$$\mathbf{v}_{surface} \cdot \mathbf{n} = \mathbf{v}_{fluid} \cdot \mathbf{n} \tag{5.3}$$

$$\mathbf{v}_{surface} \cdot \mathbf{t} = 0 \tag{5.4}$$

where $\mathbf{v}_{surface}$ is the velocity of the free surface and $\mathbf{v}_{fluid}$ is the velocity of the fluid at the free surface. These conditions assure that the motion of the free surface is equal to the motion of the fluid normal to the surface only. To help reason this out, imagine the surface to be a thin film spread out across the surface of a fluid bath. If the fluid is viewed at a perspective tangent to the surface in an Eulerian frame of reference, only motion of the fluid against the film will produce noticeable motion in the film, perhaps fluid being added to or drained from the bath to raise or lower the height of the film, or a wave passing under the film, causing it to rise or lower locally. Viewing the film from this tangential perspective would not allow observation of the fluid dragging the film but only observation of the fluid either pushing or pulling the film.

To describe the motion of the free-surface, the Monge surface consisting of the profile of the deformable region of the computational domain is used. A Monge surface is a surface that is generated by a profile curve whose profile rolls without slipping over a developing surface. To describe the evolution of any point on this surface, it is necessary

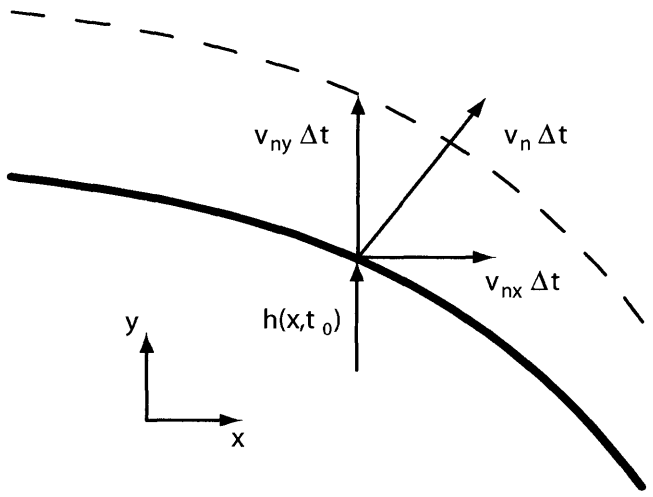


Figure 5-1: Schematic diagram of the change in the height of the surface after time Δt as described by the y component of the normal velocity of the fluid. $(-)$: surface at $t = t_0$; $(- -)$: surface at $t = t_0 + \Delta t$.

to transform eq. 5.3 into an equation describing the time rate of change of the height of the surface. The shape of the Monge surface is given in the Eulerian frame of reference by describing the height of the surface as a function of position and time. The primary direction of flow is defined as x direction, and the transverse direction of flow is defined as the y direction. The time rate of change of the surface height, dh/dt, can then be expressed as the y component of the velocity of the fluid normal to the surface, $\mathbf{v}_{ny}$ (eq. 5.5).

$$\frac{dh\,(x,t)}{dt} = v_{ny}\,(x,t) \tag{5.5}$$

The motion after a time Δt is shown pictorially in figure 5-1. This is not, however, a complete description of the time rate of change of the height of the surface, which can be seen visually in that the location of the surface does not equal the distance the fluid pushing on the surface would have travelled, $\mathbf{v}_n \Delta t$. When considering a wave travelling across the surface, it is clear that viewed in an Eulerian frame of reference the surface will appear to move both tangential to the direction of propagation of the wave and normal to the direction of propagation of the wave. Thought of in this way, clearly there are

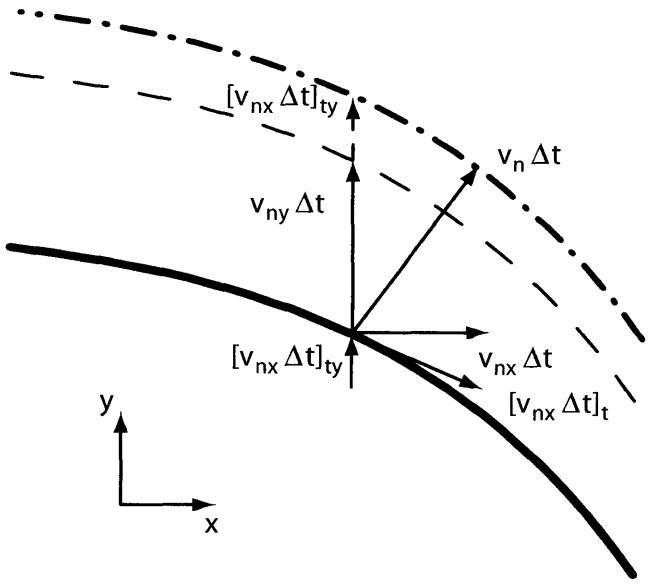


Figure 5-2: Schematic diagram of the change in the height of the surface after time Δt as described by equation 5.6. $(-)$: surface at $t = t_0$; $(- -)$: surface at $t = t_0 + \Delta t$; $(- \cdot -)$: surface at $t = t_0 + \Delta t$ accounting for translation.

contributions to the change of the height of the surface from both the local fluid velocity pushing the perpendicular direction and the motion of the wave itself. To account for this additional motion, the portion of the normal fluid velocity that would cause pure translation of the surface is included in the expression for the time rate of change of the height of the surface (eq. 5.6).

$$\frac{dh\,(x,t)}{dt} = v_{ny}\,(x,t) + \left(\frac{t_y}{t_x}\right) v_{nx}\,(x,t) \tag{5.6}$$

Here, $\mathbf{t}$ is a unit vector tangent to the surface. The motion after a time Δt considering this additional contribution is shown pictorially in figure 5-2. Visual inspection of figure 5-2 verifies that the motion of the surface over a timestep Δt now moves the full $\mathbf{v}_n \Delta t$ as described by the normal kinematic boundary condition (eq. 5.3).

5.1.3 Mesh Mapping Equations

Elliptic mapping equations are used to describe the node locations of the deformable regions of the computational domain. They are based on those derived by Christodoulou and Scriven [21], which are a modified version of the conformal mapping method

$$\nabla^2 \xi = 0 \tag{5.7}$$

$$\nabla^2 \eta = 0 \tag{5.8}$$

where ξ and η are the coordinates of the element nodes in the computational domain. Here the Laplace equations are used to ensure smooth, orthogonal meshes. An orthogonal mesh can also be created by including a factor that stretches the mesh by the element aspect ratio, S.

$$\nabla \cdot S \nabla \xi = 0 \tag{5.9}$$

$$\nabla \cdot \frac{1}{S} \nabla \eta = 0 \tag{5.10}$$

$$S = \sqrt{\frac{x_\xi^2 + y_\xi^2}{x_\eta^2 + y_\eta^2}} \tag{5.11}$$

Here , $x_\xi \equiv \partial x / \partial \xi$, etc.. Smoothness of the mesh is accounted for by a weighted version of the conformal mapping equations.

$$\epsilon_s \nabla^2 \xi = 0 \tag{5.12}$$

$$\epsilon_s \nabla^2 \eta = 0 \tag{5.13}$$

Here the parameter ϵ_s scales the relative importance of smoothness to orthogonality in the mesh in the final form of the mapping equations, typically taking on a value around

1, though higher values can be used. The composite of eqs. 5.9 and 5.12 yields eq. 5.14.

$$\nabla \cdot \left(\sqrt{\frac{x_\xi^2 + y_\xi^2}{x_\eta^2 + y_\eta^2}} + \epsilon_s \right) \nabla \xi = 0 \tag{5.14}$$

Due to the nature of the meshes generated in the die swell problem where large aspect ratios dominate in the principal flow direction, use of the stretch factor, S, in the primary flow direction is not necessary. Instead, the Laplace equation for this direction is rewritten to account for the importance of smoothness and orthogonality in the principal flow direction, x, relative to the transverse flow direction, y, by use of the scaling ratio ϵ_x/ϵ_y.

$$\nabla \cdot \left(\frac{\epsilon_x}{\epsilon_y} \eta_x \mathbf{e}_x + \eta_y \mathbf{e}_y \right) = 0 \tag{5.15}$$

Eqs. 5.14 and 5.15 are the equations used to update the element node locations at each timestep. The mapping equations in this form were first used by Smith [72].

5.1.4 Numerical Methods

A mixed finite element method is used to discretize the governing equations. Linear basis functions are used to represent the pressure, and stress unknowns, while quadratic basis functions are used to represent the velocity and nodal location unknowns. Quadrilateral elements are used to discretize the physical geometry.

The interest here is in finding the steady-state solution to the problem described by the governing equations in Secion 5.1.1. To accomplish this, the time-dependent versions of the governing equations are integrated until a steady-state solution is reached. A decoupled equation technique based on the technique described in Section 3.3 is used to solve the equations. In addition to the subsets of equations describing the polymer flow field and the polymer stress, a subset of equations describing the evolution of the free surface and element node locations is now included (fig. 5-3). The overall calculation of the evolution during a single timestep now consists of first calculation of the velocity

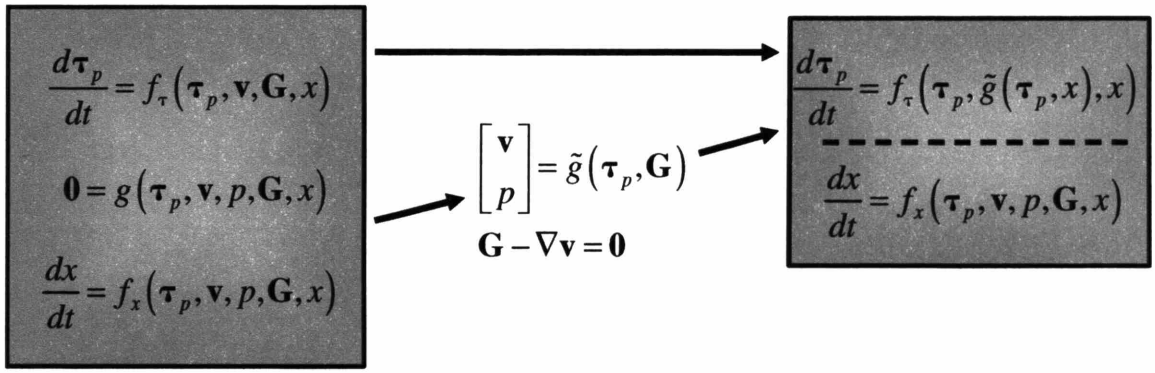


Figure 5-3: Decoupling of equation set describing the time-dependent viscoelastic flow problem with free-surface boundaries.

and pressure from the momentum and continuity equations, next the polymer stress is updated based on the new flow field, and finally the free surface is evolved and the element node locations recomputed based on the new flow field. Figure 5-4 shows schematically the progression of steps in the method. The arrows between each subproblem block not only show the progression of the solver but also denote the flow of information between each subproblem.

As with the confined flow problem, the flow equations in the DEVSS-G formulation are solved using the Harwell frontal solver. The stress equations are time integrated using the fourth order Runge-Kutta method, an explicit time integrator that allows full decoupling of the stress equations from the flow equations at each time step. An explicit integration method is also used in time integration of the free-surface boundary condition (eq. 5.6) to take advantage of the decoupling at each time step. It is important to understand the relative stability of the free surface boundary condition with respect to the stress equations. To determine best what order of accuracy is needed in the time integration method for eq. 5.6 the effects of three different explicit time integration routines used for evolution of the free-surface boundary condition were studied, namely the forward Euler method, the second order Runge-Kutta method, and the fourth order Runge-Kutta method. Results of this study showed that the evolution equation describing the free-

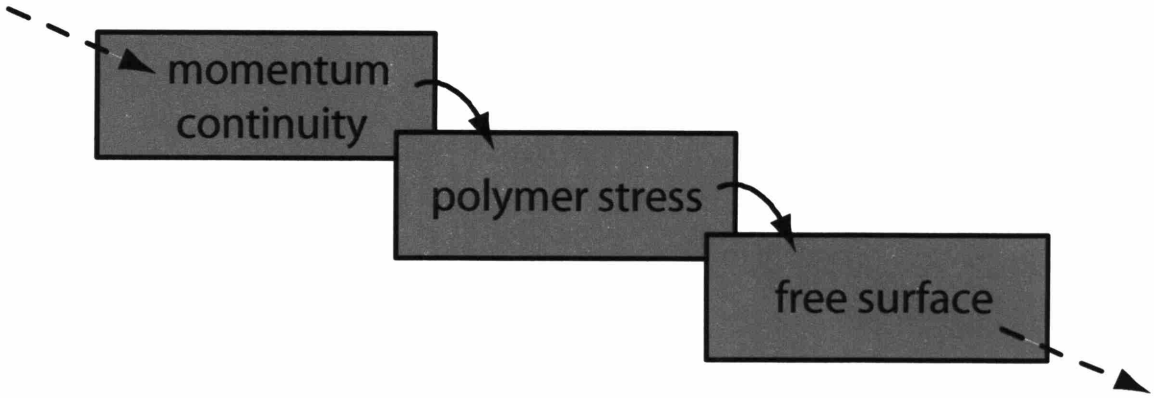
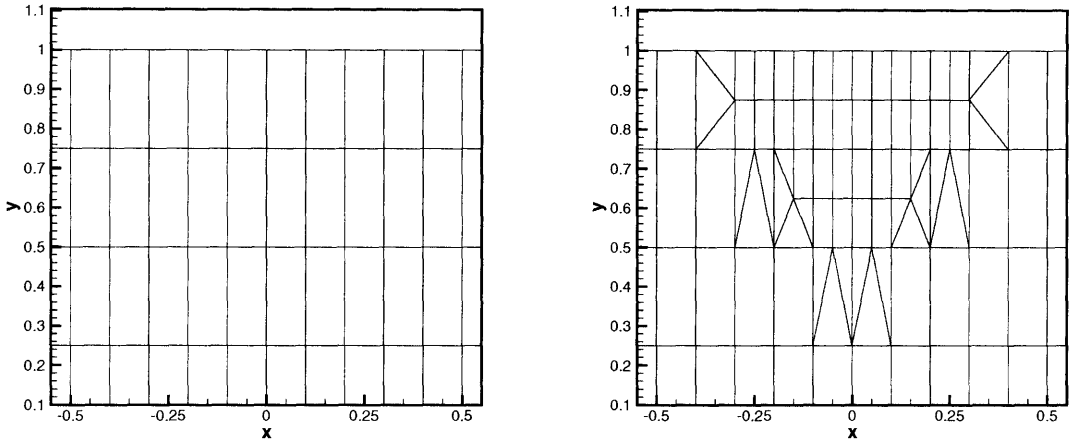


Figure 5-4: Subset steps in a single timestep for the free-surface time integrator. Arrows indicate both progression of algorithm as well as flow of information between subproblems.

surface height was time stable in the viscoelastic equation set when using the forward Euler method, a promising result considering the subset of equations describing the free-surface and element node locations would only have to be updated once for each timestep taken in the overall timestepper, while the stress equations and flow field equations are solved for in four separate substeps. To solve for the element node locations in the deformable region of the mesh, the Harwell frontal solver is employed with the node locations of the boundaries of the deformable region at the current time step acting as essential boundary conditions.

Judgement of steady state is made through comparison of the L_2 norms of the velocity fields and stress fields from one timestep to the next. When the difference between timesteps in both norms drops below a given tolerance, the method is deemed to have reached the steady state for the problem.

Since the method detailed herein is a new formulation for the viscoelastic problem, it is important to analyze the convergence of the method with successive mesh refinement. Figure 5-5 shows the successive levels of refinement in the region surrounding the contraction die lip. As shown in [67] refinement around the singularity is necessary to capture correctly the stress as the fluid dettaches from the die. Here, convergence of



(a) M1

(b) M2

(c) M3

(d) M4

Figure 5-5: Successive levels of mesh refinement in region surrounding the die lip. Die lip located at $x = 0$, $y = 1$.

a macroscopic quantity of the flow will be used to show that the method is convergent, namely the die-swell ratio, $\epsilon = h_f/h - 1$. The plot of die-swell ratio as a function of mesh refinement near the die lip, figure 5-6, shows h^2 convergence of the method.

5.2 Die-swell of a Giesekus fluid as a function of die aspect ratio

In order to test the accuracy of the time-dependent method to known solutions, simulation of planar die-swell from a contraction die was performed for a Giesekus fluid with mobility parameter $\alpha = 0.2$ and solvent to total viscosity ratio $\beta = 0.5$. The geometry is shown in figure 5-7. Flow through the geometry is from left to right. For the flow problem, on the inflow boundary the one-dimensional version of the flow equation is solved for the x component of velocity and the y component of velocity is set to zero. On the outflow boundary, the normal stress is specified to be zero in the x direction and the y component of velocity is again set to zero. On the walls both the x and y components of velocity are set to zero. On the centerline boundary, a symmetry condition is used for the x component of velocity and the y component of velocity is set to zero. On the meniscus boundary, eqs. 5.1 and 5.2 are specified as natural boundary conditions. For the stress problem, since the evolutino equations are hyperbolic in nature, only a single boundary condition can be used on the inflow boundary. For this condition, the one-dimensional versions of the stress equations are solved on the inflow boundary. For the deformable mesh problem, essential conditions of the node locations are specified on all boundaries.

Simulation results using the time-dependent method are compared to solutions from the steady-state method used in [72]. For these simulations, the Weisenberg number, a measure of viscoelasticity in the system, is defined as $\mathrm{We} \equiv \lambda \langle v \rangle / h$ where λ is the polymer relaxation time, $\langle v \rangle$ is the average velocity in the die land, and h is the half height of the die land. The die-swell ratio, the percent swell of the free-surface beyond

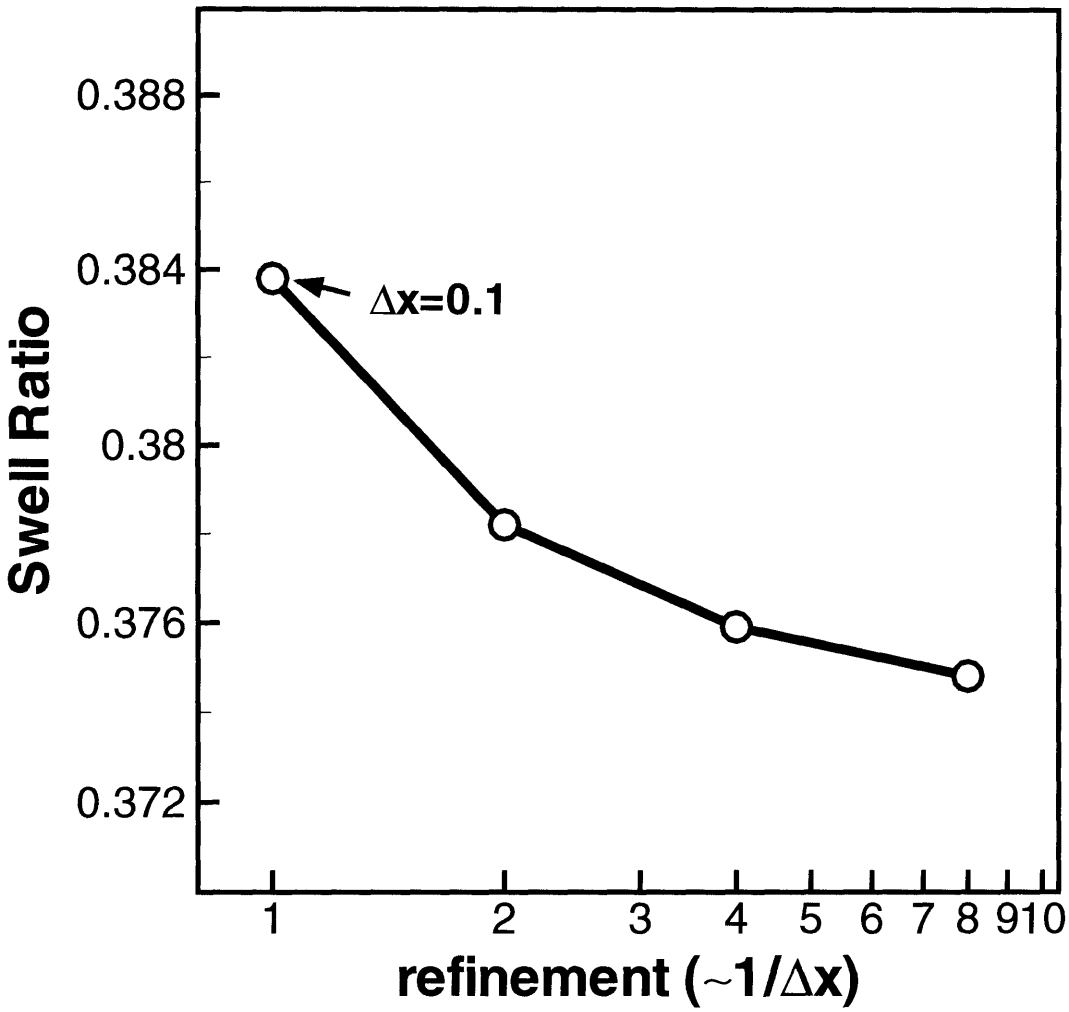


Figure 5-6: Die-swell ratio versus level of mesh refinement around die lip for flow of a Giesekus fluid ($\alpha = 0.2$, $\beta = 0.5$) through a contraction die-swell geometry.

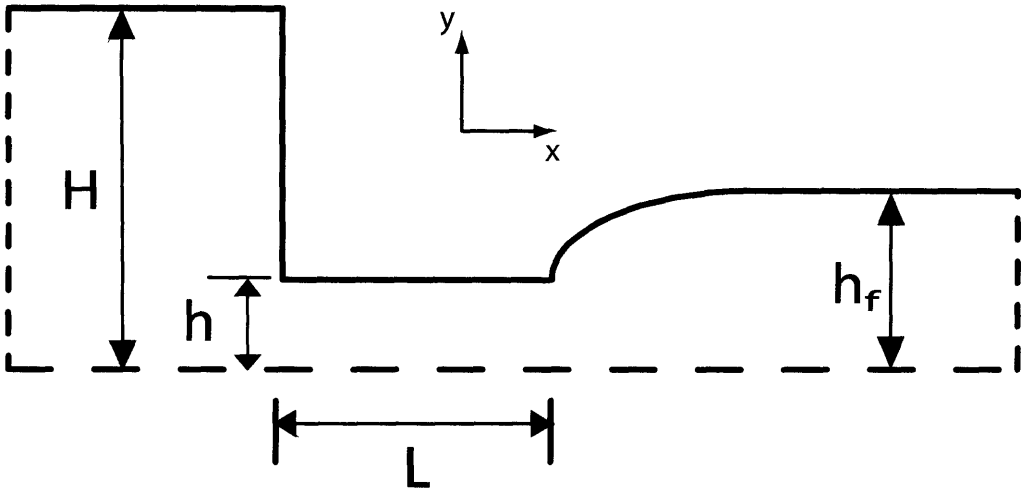


Figure 5-7: Planar die-swell geometry. H: upstream height; h: contractoin height; L: contraction length; h_f: maximum swell height.

the die half height, is defined as $\epsilon = h_f/h - 1$ where h_f is the maximum half height of the polymer surface downstream of the die. In the following simulations the contraction ratio is fixed at $H/h = 4$, while the die land length, L/h, and the capillary number, Ca, are varied over the range of the parameter space.

The mesh used for the bulk of the simulations is shown in Figure 5-8. The mesh

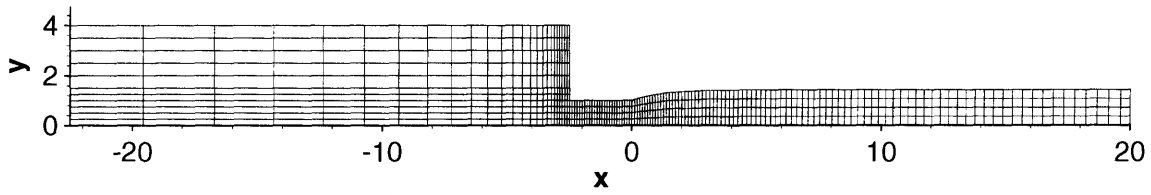


Figure 5-8: Contraction die-swell mesh used for comparisons of the time-dependent and steady-state free-surface solvers.

consists entirely of quadrilateral elements, with a concentration of elements in the die land as well as just upstream and downstream of the die land.

Contour plots for the kinematic variables for a We = 1 flow are shown in Figure 5-9. The majority of the pressure drop for this flow is seen in the die land, with an increased

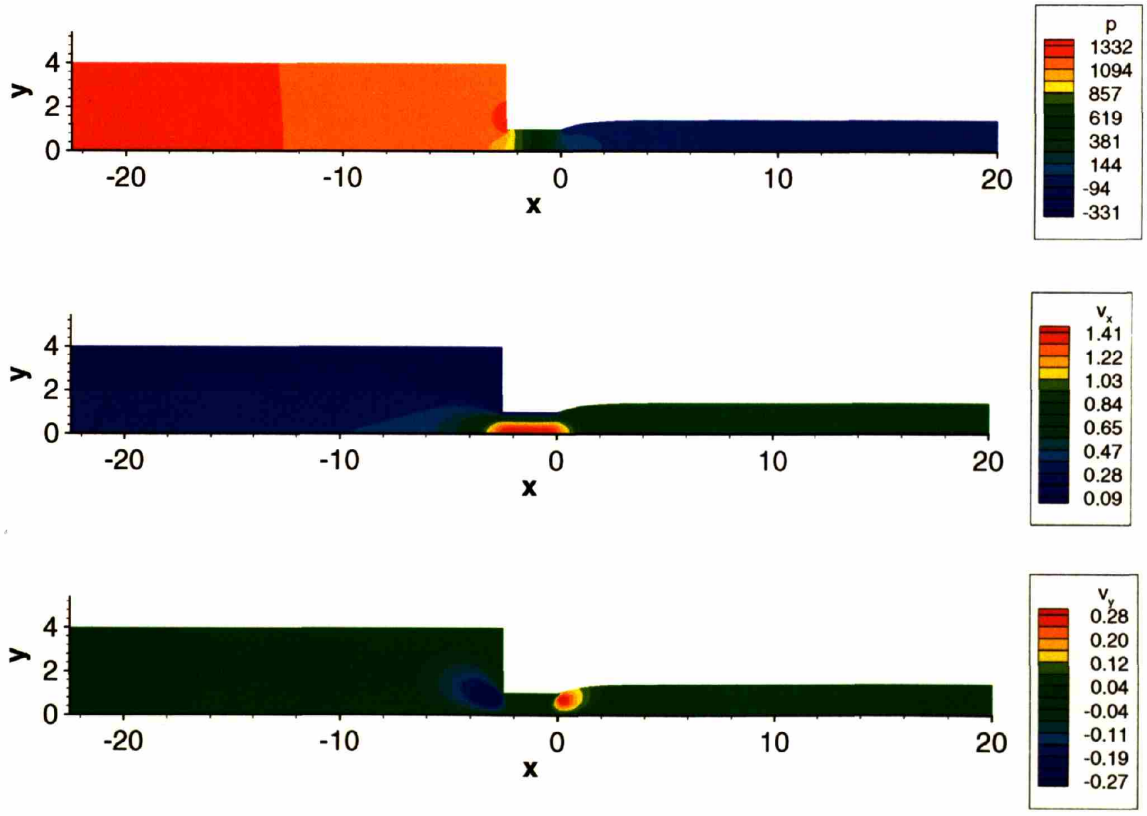


Figure 5-9: Contour plots of pressure and velocity for flow through the contraction die-swell geometry with We = 1, $L/h = 2.5$, and $Ca = 1$ for the Giesekus model ($\alpha = 0.2$ and $\beta = 0.5$) as computed with the time-dependent method.

pressure region on the upstream side of the reentrant corner of the contraction and a negative pressure spike at the die lip. The velocity and pressure profiles continue to vary until the meniscus boundary has reached a constant height, no more than $5h$ lengths from the die exit.

Comparisons of the solutions computed using the time-dependent method and steady-state method were made over the range of the parameters pertaining to viscoelasticity, We, to surface tension, Ca, and to the geometry, L/h. For comparisons of the two

methods over the range of We, the capillary number is fixed at Ca = 1 and the die land length is fixed at $L/h = 2.5$. Figure 5-10 shows the change in the die-swell ratio over a range of We as computed with both the time-dependent and steady-state methods. As We is increased, the swell ratio of the polymer exiting the die increases monotonically. This increase is undoubtedly due to the increased buildup of normal stresses inside the die land just prior to the fluid emerging from the die as the viscoelasticity of the fluid is increased. However, it is important to note that for $L/h = 2.5$, the memory of the shape of a fluid element in the large slit plays a significant part in the resulting swell. Results from the time-dependent method show excellent agreement with the results from the steady-state method.

For comparison of the two methods over the range of L/h, the capillary number is fixed at Ca = 1 and the Weisenberg number is fixed at We = 4. Figure 5-11 shows the change in die-swell ratio over a range of L/h as computed with both the time-dependent and steady-state methods. Holding We constant for the flow as L/h is increased allows for observation of the effects of the diminishing memory of the polymer fluid as it is given greater time to relax following the narrow contraction region. As the die land length is increased, the swell ratio of the polymer decreases logarithmically, approaching a limit of approximately 0.355 as L/h approaches infinity. Results from the time-dependent method again show excellent agreement with the those of the steady-state method.

For comparison of the two methods over the range of Ca, the Weisenberg number is fixed at We = 4 and the die land length is fixed at $L/h = 2.5$. Figure 5-12 shows the change in die-swell ratio over a range of Ca as computed with both the time-dependent and steady-state methods. Since the capillary number is a measure of the relative strength of the viscous and surface forces, varying Ca while holding We and L/h constant for the flow allows for study of the effects of varying the normal forces acting on the free-surface of the polymer, which tend to constrain swell of the fluid due to surface tension. Increasing Ca effectively decreases the normal forces acting on the surface, as can be seen in the normal stress balance on the surface in eq. 5.2. This can be seen in Figure 5-12.

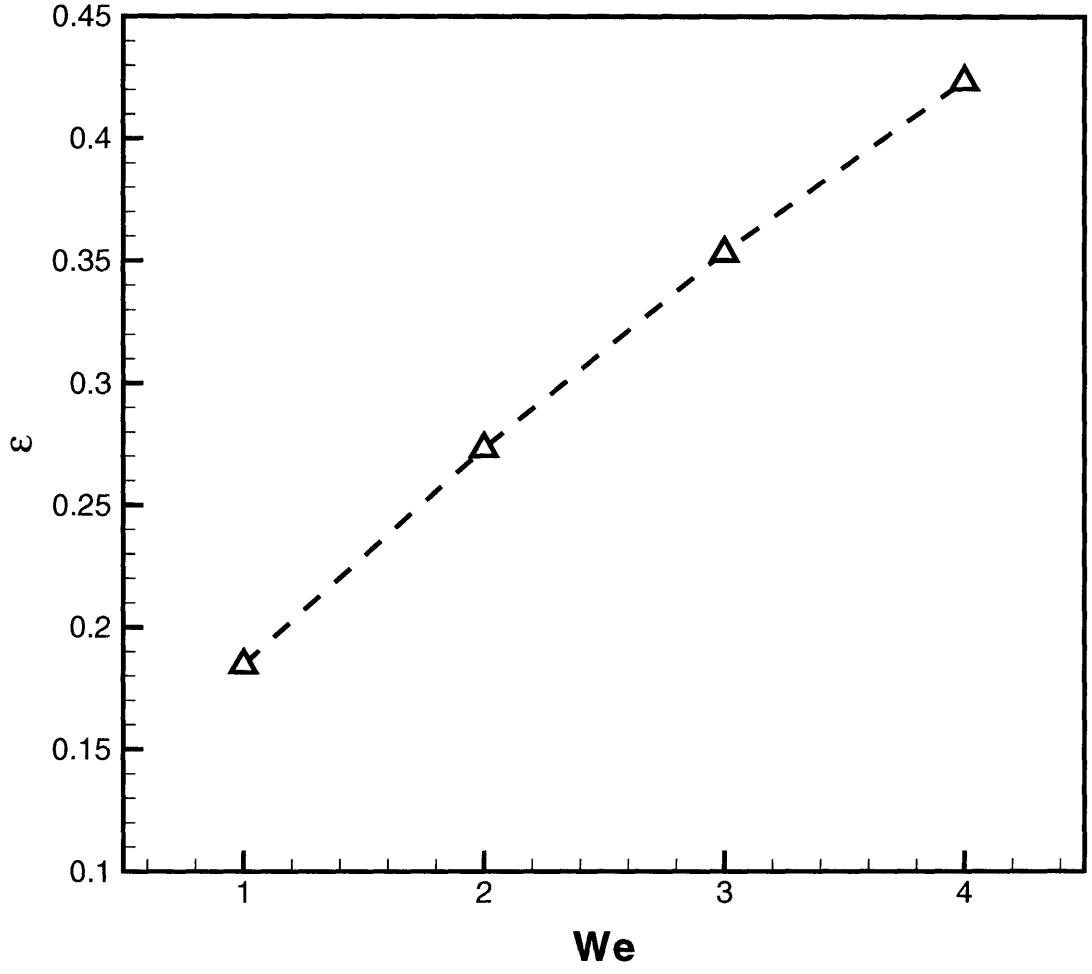


Figure 5-10: Die-swell ratio versus Weisenberg number for flow of a Giesekus model ($\alpha = 0.2$ and $\beta = 0.5$) in a contraction die-swell geometry with $L/h = 2.5$ and Ca $= 1$. ($--$): steady-state method; ($\triangle$): time-dependent method.

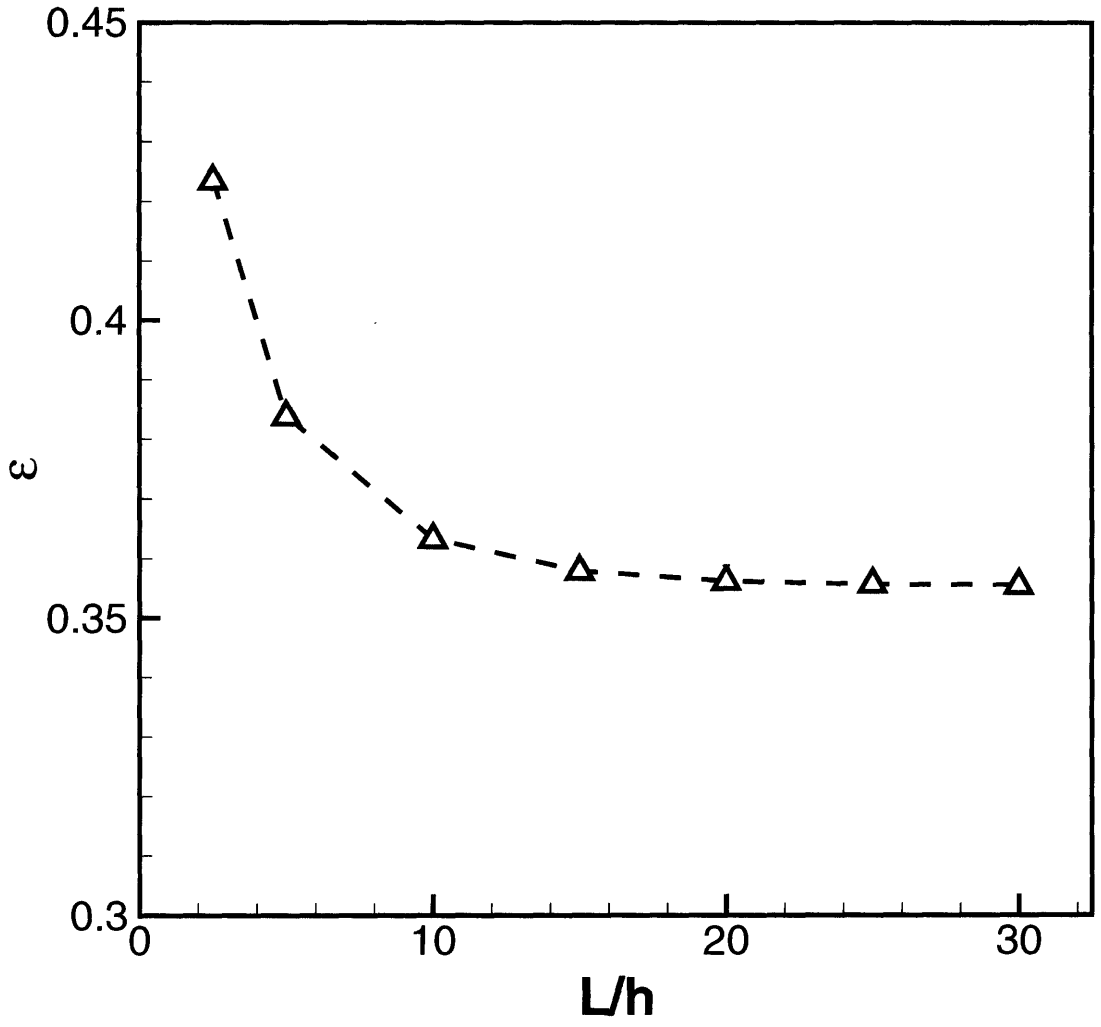


Figure 5-11: Die-swell ratio versus die land length for flow of a Giesekus model ($\alpha = 0.2$ and $\beta = 0.5$) in a contraction die-swell geometry with Wi $= 4$ and Ca $= 1$. (−−): steady-state method; (△): time-dependent method.

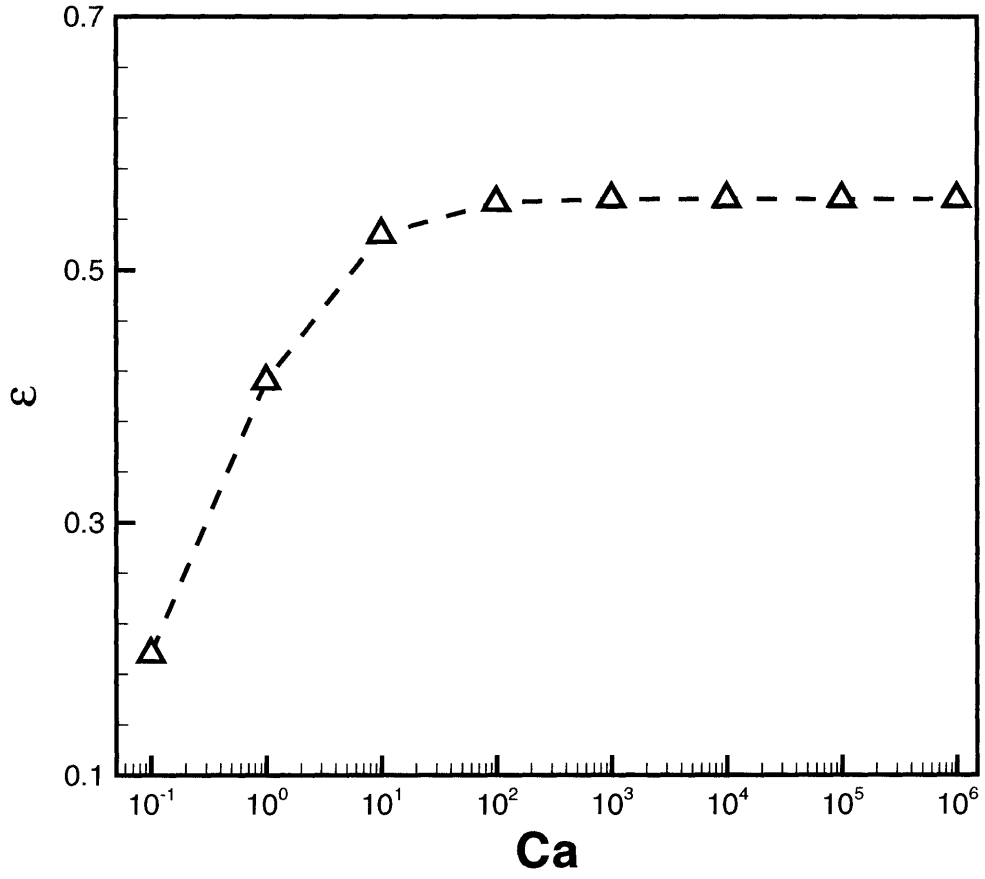


Figure 5-12: Die-swell ratio versus capillary number for flow of a Giesekus model ($\alpha = 0.2$ and $\beta = 0.5$) in a contraction die-swell geometry with We $= 4$ and $L/h = 2.5$. ($--$): steady-state method; ($\triangle$): time-dependent method.

As Ca is increased the swell ratio increases logarithmically, approaching a maximum of 0.556. It is also interesting to note that as Ca is decreased from unity the swell ratio monotonically decreases due to the singularity in the normal stress boundary condition arising as $\mathrm{Ca} \rightarrow 0$.

5.3 Conclusions

In this chapter a new finite element formulation has been discussed for solving the viscoelastic flow problem with free-surface boundary conditions. Continuing along the vein of reduction of the overall computational cost of the viscoelastic flow problem, the equations describing the updating of the element node locations within the deformable mesh region and the associated boundary conditions are included as a new subproblem within the previously-developed decoupled time-dependent method (Section 3.3). In order to implement the equations in this method, the kinematic boundary condition for the normal velocity of the meniscus is converted into an evolution equation for the height of the surface. Modified versions of the elliptical mapping equations derived by Christodoulou and Scriven [21] along with essential conditions of the locations of all nodes on the boundaries of the deformable region at the current time step are then solved to update the element node locations in the deformable region.

Comparison of the solutions of the new time-dependent method are made to a previously developed steady-state method by using the flow of the Giesekus model in a contraction die-swell geometry as a test problem. Simulation results are compared for a range of We, Ca, and L/h. All simulation results show excellent agreement with those computed using the steady-state method. Therefore, the time-dependent implementation is a viable numerical method for the computation of the viscoelastic free-surface problem in two-dimensional flow.

Chapter 6

4:1:4 Axisymmetric Contraction-expansion Flow

6.1 Background

The flow through contraction-expansion geometries is a complex combination of both shearing and extensional kinematics. Experimental work with Boger fluids in this class of geometries has revealed interesting dynamics, such as a pressure drop enhancement with increasing flow rate [[64], [65]]. Previous attempts to model this behavior have failed [[80], [22], [46]]. Inability of the polymer models to properly describe the polymer in regions of rapid extension may be responsible for these failures. Rothstein et al. [65] also identified major differences in vortex growth for various contraction ratios, namely shrinking of the upstream vortex with subsequent formation of a lip vortex in the 2:1:2 geometry. The 4:1:4 and 8:1:8 ratios both produced growing upstream vortices with increasing flow rate. Here we employ the closed version of the Adaptive Length Scale model, ALS-C, developed by Ghosh et al. [34] to describe the flow of the polymer in the 4:1:4 contraction-expansion geometry. The ALS-C model has been used previously to capture the asymmetric wake instability in the flow around a cylinder problem. We also consider the use of a multimode FENE-P model in an attempt to model the system

better.

Section 6.2 describes the physical geometry and fluid properties modeled to represent those used by Rothstein et al., as well as the governing equations and numerical methods used to simulate the system. Section 6.3 describes the rheological fit of the model parameters to the physical fluid. Section 6.4 contains the results of the simulations of flow in the 4:1:4 geometry. Section 6.4.1 contains discussion of the results obtained with the FENE-P and ALS-C models. Section 6.4.2 contains discussion of the results obtained with the 4-mode FENE-P model and compares them to the single mode model results.

6.2 Problem Description

6.2.1 Physical Geometry

The geometry considered here is an axisymmetric 4:1:4 contraction-expansion designed to match the experimental system used in [65], a schematic of which is shown in Figure 6-1. All physical dimensions are non-dimensionalized by the small-tube radius, R_2. The length of the contracted region is defined as $L_c = R_2$, and the reentrant corner is rounded with a radius of $R_c = 0.5R_2$. The downstream corner is rounded with a radius of $R_{c_2} = 0.2R_2$ to avoid a computational singularity. The tube radius upstream and downstream of the contraction is equal to four times the small-tube radius, $R_1 = 4R_2$. The upstream and downstream channel lengths, L_u and L_d respectively, are chosen to be sufficientlty long to be able to impose fully developed flow boundary conditions on the inlet and outlet boundaries. Specifically we choose $L_u = 24R_2$ and $L_d = 25R_2$, yielding an overall geometry length of $L = L_u + L_c + L_d = 50R_2$.

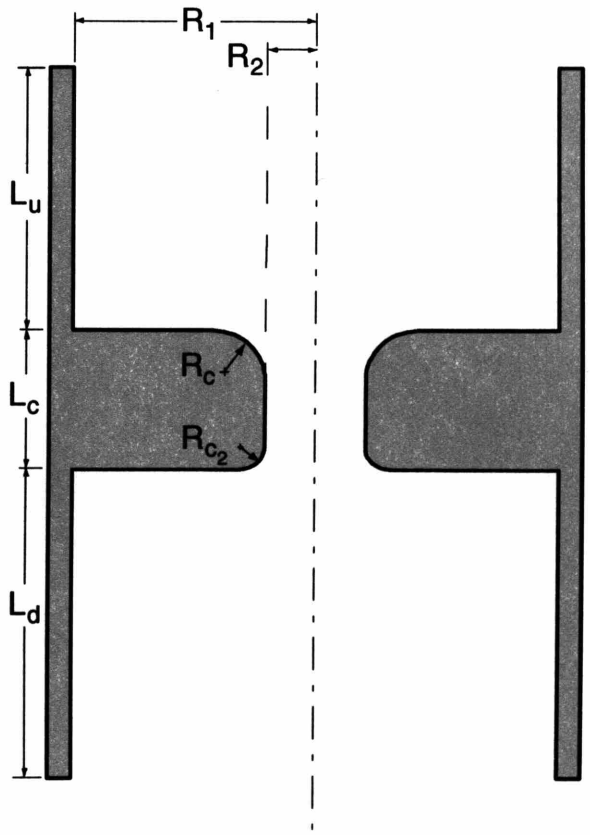


Figure 6-1: 4:1:4 contraction expansion geometry selected to model the experimental geometry of Rothstein et al. [65]. Flow through the geometry is from top to bottom. Here $R_1 = 4R_2$, $L_c = R_2$, $L_u = 24R_2$, $L_d = 25R_2$, $R_c = 0.5R_2$, and $R_{c_2} = 0.2R_2$.

6.2.2 Fluid properties

The fluid modeled here is a 0.025% by weight solution of monodisperse polystyrene dissolved in oligomeric styrene. The rheological properties of this fluid and its relevant physical parameters are reported in [65] and are summarized in Table 6.1.

Notation	Description	Parameter Value
c	Concentration of high molecular weight polystyrene	0.025%
M_w/M_n	Polydispersity	1.03
M_w	Molecular weight (g/mol)	2.0 x 10^6
$b = L^2$	Extensibility parameter	26900
T_0	Reference temperature (K)	298
η_0	Zero-shear-rate viscosity (Pa s)	22.75
η_s	Solvent viscosity (Pa s)	21
λ_{ps}	Solvent relaxation time (s)	2.5 x 10^{-4}
λ_Z	Zimm (longest) relaxation time (s)	3.24
$\overline{\lambda}$	Oldroyd relaxation time (s)	0.146
Ψ_{10}	Zero-shear-rate first normal stress coefficient (Pa s^2)	6.66

Table 6.1: Parameters characterizing the viscometric properties of the 0.025 wt.% PS/PS fluid as reported by Rothstein et al. [2001]

6.2.3 Modeling and Simulation

Modeling of the rheological properties of the fluid for determination of the ALS-C and FENE-P model parameters was carried out with a homogeneous flow simulation of both steady shear and transient extensional flow developed in the Matlab environment.

To model the heterogeneous complex flow, the DEVSS-G formulation for the momentum and mass continuity equations is utilized [54]. For the polymer extra stress, a closed version of the Adaptive Length Scale model (ALS-C) is chosen [34], and the FENE-P model is used for comparison.

Flow of the fluid through the contraction-expansion geometry is governed by the equations of momentum, mass continuity, and equations describing the relationship between polymer stress and velocity of the fluid. In the DEVSS-G formulation, the momentum equation is written using stress-splitting to take the form

$$\nabla^2 \mathbf{v} - \nabla p - \nabla \cdot \left\{ (1-\beta) \left\{ \mathbf{G} + \mathbf{G}^T \right\} \right\} - \nabla \cdot \boldsymbol{\tau}_p = \mathbf{0} \tag{6.1}$$

Here $\mathbf{v}$ is the fluid velocity, β is the solvent to zero shear rate viscosity ratio, p is the pressure, $\mathbf{G}$ is the velocity gradient tensor, $\boldsymbol{\tau}_p$ is the extra stress, and ∇ is the gradient operator. The mass continuity equation is given by,

$$\nabla \cdot \mathbf{v} = \mathbf{0} \tag{6.2}$$

A least squares minimization is used to compute the velocity gradient,

$$(\nabla \mathbf{v} - \mathbf{G})^2 = \mathbf{0} \tag{6.3}$$

The extra stress of the polymer is described through a few different sets of constitutive equations in this work. The closed version of the Adaptive Length Scale (ALS-C) model [34] is primarily used in this work. The theory behind this model is discussed at length in Ghosh et al. [34]. For completeness, the theory of the Adaptive-Length-Scale, ALS, model as an adaptive version of the Finitely Extensible Nonlinear Elastic, FENE, model is summarized here.

The force law for the FENE model is given by

$$F^{FENE}(Q) = \frac{HQ}{1 - (Q/Q_0)^2} \tag{6.4}$$

where Q is the length of the FENE spring, Q_0 is the maximum extension of the spring, and H is the spring constant. Ghosh et al. [34] argued that while a FENE chain with an increasing number of links can be used to capture with reasonable accuracy the behavior in uniaxial extension of the more exact and quite more computationally expensive Kramers chain model, the FENE chain is not an efficient model for this task. Whereas a large number of springs are needed in the FENE chain model to capture the behavior of the Kramers chain at low strains, a much simpler FENE dumbbell model

(single spring) captures the behavior at moderate to high strains.

As a more efficient alternative, Ghosh et al. proposed a model in which a single spring has a length scale L^* that adapts to the kinematic history of the flow, the ALS model. L^* is the largest length scale for which the FENE force law approximation applies. The polymer is then represented by a FENE spring whose maximum length equals L^*. The force law of a segment with a maximum length L^* is given by

$$F_{seg}^{*(s)} = \frac{H_{seg}^* Q}{1 - (Q^2/L^{*2})} \tag{6.5}$$

where H_{seg}^* is the spring constant that is proportional to the number of segments in the spring. The number of segments is set equal to the contour length of the polymer molecule divided by L^*. L^* would be the length scale if the polymer molecule were able to react instantaneously to the surrounding environment. Ghosh et al. proposed an evolution equation for the actual length scale L that takes into account the relaxation of the polymer molecule to the flow as well as a stiffening of the polymer molecule. The evolution equation for the adaptive length is shown here in dimensionless form as

$$\frac{db_{seg}}{d\hat{t}} = -\left|\mathbf{n} \times \frac{\hat{\mathbf{Q}}}{\left|\hat{\mathbf{Q}}\right|}\right| \mathrm{Wi} \left(1 - \frac{\hat{Q}^2}{b_{seg}^*} \frac{b_{\max}}{b_{seg}^*}\right) \left(b_{seg} - b_{seg}^*\right) K^* + \left(b_{\max} - b_{seg}\right) K \tag{6.6}$$

where b_{seg} is the dimensionless adaptive length, $\mathbf{n}$ is a unit vector parallel to the eigenvector of the rate-of-strain tensor that corresponds to its largest eigenvalue, $\mathrm{Wi} = \frac{1}{2}\lambda_d \max\left[\mathrm{Eig}\left(\dot{\boldsymbol{\gamma}}\right)\right]$ is the Weissenberg number, λ_d is the longest relaxation time of the polymer, $\dot{\boldsymbol{\gamma}}$ is the rate-of-strain tensor, b_{seg}^* is the dimensionless adaptive length scale if the polymer were able to instantaneously react to the flow, and K^* and K are ratios of λ_d to the time constants (λ^* and λ) of a single segment.

Ghosh et al. demonstrated that the new model has much better agreement with the Kramers chain model than the FENE dumbbell model in a wide range of flows. In addition, they proposed a closed version of the model that could be readily used in

complex flow simulations and showed that the rheological properties of the model were virtually unchanged by the closure approximation.

In the complex flow simulations of the present work, the closed version of the ALS model developed in [34] is employed. The extra stress of the polymer in the ALS-C model is related to the average conformation of the polymer molecules by

$$\boldsymbol{\tau}_p = \frac{b_{\max}}{\langle b_{seg} \rangle}\boldsymbol{\delta} - \left(\frac{b_{\max}}{\langle b_{seg} \rangle}\right)^2 \frac{\langle \mathbf{QQ} \rangle}{1 - \frac{tr\langle \mathbf{QQ} \rangle}{\langle b_{seg} \rangle}\frac{b_{\max}}{\langle b_{seg} \rangle}} \tag{6.7}$$

Here $b_{\max}$ is the maximum extensibility of the polymer molecule, $\langle b_{seg} \rangle$ is the dimensionless finite extensibility for a segment with maximum extension equal to the adaptive length scale L, and $\langle \mathbf{QQ} \rangle$ is the conformation tensor of the molecule. $\langle b_{seg} \rangle$ is described by the evolution equation

$$\frac{D\langle b_{seg} \rangle}{Dt} = -\left|\mathbf{n} \times \begin{pmatrix} \langle Q_1 Q_1 \rangle^{1/2} \\ \langle Q_2 Q_2 \rangle^{1/2} \\ \langle Q_3 Q_3 \rangle^{1/2} \end{pmatrix} \frac{1}{(tr\langle \mathbf{QQ} \rangle)^{1/2}}\right| De \left(1 - \frac{tr\langle \mathbf{QQ} \rangle}{\langle b^*_{seg} \rangle}\frac{b_{\max}}{\langle b^*_{seg} \rangle}\right) \tag{6.8}$$
$$\left(\langle b_{seg} \rangle - \langle b^*_{seg} \rangle\right) K^* + \left(b_{\max} - \langle b_{seg} \rangle\right) K$$

Here $D\langle b_{seg} \rangle / Dt \equiv d\langle b_{seg} \rangle / dt + \mathbf{v} \cdot \nabla \langle b_{seg} \rangle$ is the substantial derivative, $\mathbf{n}$ is a unit vector parallel to the eigenvector of the rate-of-strain tensor corresponding to the largest eigenvalue, and $\langle b^*_{seg} \rangle$ is the average adaptive length of the molecule if the parameters describing it were able to instantaneously change as a result of instantaneous changes in the flow field and is given by the relation,

$$\frac{De}{K^*}\left(1 - \frac{tr\langle \mathbf{QQ} \rangle}{\langle b^*_{seg} \rangle}\frac{b_{\max}}{\langle b^*_{seg} \rangle}\right) = z \tag{6.9}$$

In eq. 6.9 z is an unknown constant of $O(1)$. K and K^* are given by the following

equation in which b is equal to $\langle b_{seg} \rangle$ and $\langle b^*_{seg} \rangle$, respectively:

$$K = \left(\frac{b}{b+5}\right)\left(\frac{\sqrt{(b_{\max}+5)(b_{\max}+7)}}{b_{\max}}\right)\cdots \qquad (6.10)$$
$$\left(\frac{\left(2(M+1)^2+7\right)\left((M+1)^2-1\right)}{45} - \frac{12\left((M+1)^2+1\right)\left((M+1)^2-1\right)}{45(M+1)(b+7)}\right)^{1/2}$$

Here $M = b_{\max}/b$ is the average number of segments in the molecule. $\langle \mathbf{QQ} \rangle$ is given by the evolution equation,

$$\frac{D\langle \mathbf{QQ} \rangle}{Dt} = De\left\{\nabla \mathbf{v}^T \cdot \langle \mathbf{QQ} \rangle + \langle \mathbf{QQ} \rangle \cdot \nabla \mathbf{v}\right\} - \frac{K\langle \mathbf{QQ} \rangle}{\left(1 - \frac{tr\langle \mathbf{QQ} \rangle}{\langle b_{seg} \rangle}\frac{b_{\max}}{\langle b_{seg} \rangle}\right)} + K\left(\frac{\langle b_{seg} \rangle}{b_{\max}}\right)\boldsymbol{\delta} \quad (6.11)$$

Motivated by the fact that the ALS model is essentially a multimode FENE model with a spectrum of time constants rather than individual discrete modes, single mode and four mode FENE-P models are also employed for comparison to the ALS-C model. For the FENE-P model, the extra stress of the polymer is related to the average conformation of the polymer molecules by

$$\boldsymbol{\tau}_p = \left(\frac{b+5}{b+2}\right)\boldsymbol{\delta} - \frac{\langle \mathbf{QQ} \rangle}{1 - \frac{tr\langle \mathbf{QQ} \rangle}{b}} \qquad (6.12)$$

where $\langle \mathbf{QQ} \rangle$ is given by the evolution equation

$$\frac{D\langle \mathbf{QQ} \rangle}{Dt} = De\left\{\nabla \mathbf{v}^T \cdot \langle \mathbf{QQ} \rangle + \langle \mathbf{QQ} \rangle \cdot \nabla \mathbf{v}\right\} - \frac{\langle \mathbf{QQ} \rangle}{\left(1 - \frac{tr\langle \mathbf{QQ} \rangle}{b}\right)} + \left(\frac{b+5}{b+2}\right)\boldsymbol{\delta} \qquad (6.13)$$

On the inflow boundary the velocity in the axial flow direction is specified through solution of the one-dimensional kinematic equations, and the velocity in the radial direction is specified as zero. No slip boundary conditions are specified on all of the solid walls in the geometry. On the centerline no variation in the radial direction of the axial velocity and zero radial velocity are specified. On the outflow boundary a fully developed

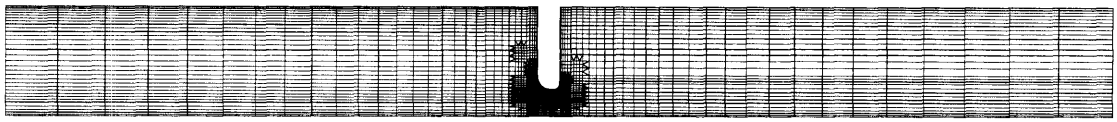


Figure 6-2: Finite element mesh necessary to discretize 4:1:4 contraction-expansion geometry.

condition is applied to the axial velocity, and the radial velocity is specified as zero. There are assumed to be no variations in the azimuthal direction, and the azimuthal velocity is specified as zero. Boundary conditions for the conformation tensor and adaptive length scale are set on the inflow boundary through the solution of one-dimensional versions of the respective evolution equations.

The Galerkin finite element method is used to discretize the unknowns for the DEVSS-G equations, namely $\mathbf{v}$, p, and $\mathbf{G}$, and the discontinuous Galerkin method is used for unknowns for the constitutive equations describing the polymer stress, namely $\langle \mathbf{QQ} \rangle$ and b_{seg}. The mesh used to discretize the physical domain is shown in figure 6-2. This structured mesh combines both quadralaterals and triangles for its constituent elements. Whereas quadralaterals are used for the bulk of the mesh, triangles are used to match boundaries between regions of differing levels of refinement. By employing triangles in this manner, use of psuedo-node formulations can be avoided without the cost of refinement in regions where it is unnecessary to represent the flow optimally. Figure 6-3 shows the amount of refinement used near the contraction wall, where the gradients of the velocity, velocity gradients, and stress are highest. The smallest element in this mesh is $0.01R_2$ by $0.01R_2$, located next to the wall in the contraction throat.

The index one set of DAE's resulting from the finite element discretization is then converted into a set of first order ODE's by using the operator splitting technique described in [77]. Once written in this form, the set of equations can be decoupled and solved in separate sub-steps by employing an explicit time integration technique, in which

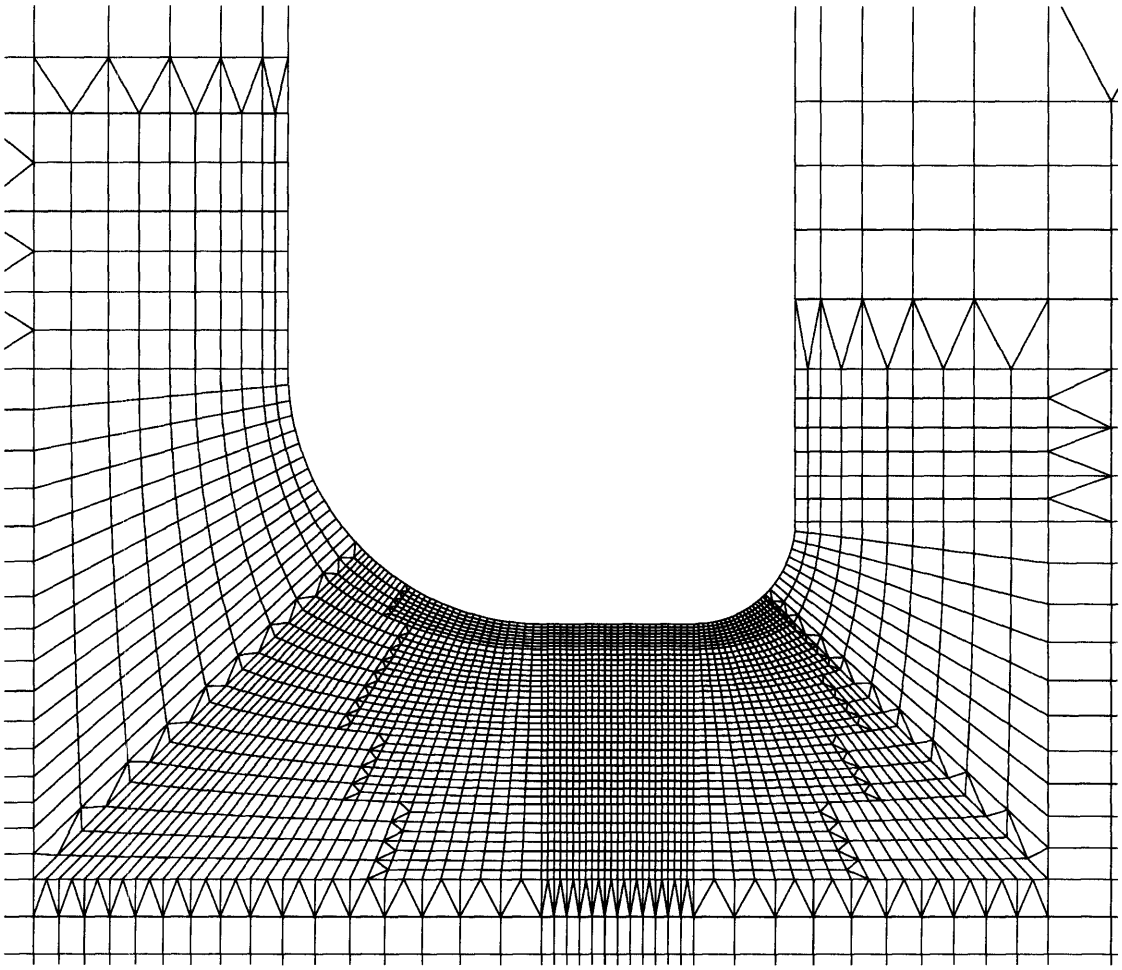


Figure 6-3: Level of refinement necessary near the contraction wall to resolve areas of high gradients.

the equations describing the flow, equations 6.1, 6.2, and 6.3, become constraints on the evolution equations describing the polymer extra stress, equations 6.11 and 6.8. The flow equations are further decoupled by first solving equations 6.1 and 6.2 and then using the new velocity fields to compute the **G** interpolants from equation 6.3. The momentum and continuity equations are preconditioned and then solved with a Krylov iterative technique as in [16]. Unlike the work of Caola et al. [16] in which BiCGStab was used as the iterative solver, here the GMRES algorithm is employed for its improved stability and robustness. Also unlike the work of Caola et al., a separate solution technique can now be employed to solve for the least squares interpolation for **G** since equation 6.3 is now decoupled from the flow equations. Here the Conjugate Gradient method is employed for this symmetric, positive definite set of equations. All of the sub-problems are solved by using parallel versions of the Krylov solvers and the domain decomposition CHACO to optimally distribute the equations among the nodes in the parallel machine. Complete details of this method are discussed in [16]. The computations were carried out on a parallel cluster of 32 Dell 2650 PowerEdge servers each with dual 2.8 GHz Intel Pentium Xeon processors and each with one gigabyte of memory. The intercommunication between machines was carried out via an Extreme Black Diamond 6808 switch with gigabit copper NIC's and adequate bandwidth for a fully non-blocking and non-oversubscribed architecture.

6.3 Fluid rheology and model parameter determination

Model parameters for the ALS-C and FENE-P constitutive equations are determined by attempting to match closely the rheology of the model to that reported in [65]. The zero-shear-rate Deborah number, $De = \lambda_0 \langle v_z \rangle_2 / R_2$, is used here to present the data. The zero-shear-rate relaxation time, $\lambda_0 = \Psi_{10}/2\eta_{p0}$, is calculated to be 1.886 seconds. The single mode fit to the complex viscosity data from small angle oscillitory shear flow

reported in [65] is shown in Figs. 6-4 and 6-5. As is apparent from the plots of η' and η'', the single mode fails to capture the decrease in the dynamic rigidity with increasing frequency. The solvent mode is assumed to behave Newtonian-like over the range of experimentally accessible shear rates and frequencies.

A four mode FENE-P model was also fit to the rheological data. Parameters for each mode of the four mode FENE-P model are given in Table 6.2.

mode	$\boldsymbol{\lambda}_k$	$\boldsymbol{\eta}_{pk}$	b
1	4.75	0.290	30
2	1.25	0.387	300
3	0.5	0.129	115
4	0.05	0.194	50

Table 6.2: Parameters for the 4 mode FENE-P model

Choice of the time constant, λ_k, for each mode was made to ensure the modes were well distributed over the range of response demonstrated in the linear rheological data. The contribution of each mode to the total polymer viscosity was then determined by fit to the linear viscoelastic data, results of which are shown in Figs. 6-6 and 6-7. Through use of the additional modes, much better agreement with the linear viscoelastic data is acheived.

For the single-mode models a maximum extension of $b = 7,744$ is used to fit the steady-state extensional viscosity. Choice of the z parameter for the ALS-C model is best determined from comparison of the transient extensional viscosity of the model and experiments. Figure 6-8 compares the transient extensional viscosity for the ALS-C model, FENE-P model, and experimental data. Here, the FENE-P model prediction lags the data in strain; in contrast, the ALS-C model more closely captures the strain dependence of the extensional viscosity. The steady-state extensional viscosity of the fluid is captured by both models. The first normal stress coefficient in steady shear flow for the ALS-C and FENE-P models is shown in Figure 6-9. Although the ALS-C model for all z values is in the same range as the experimental data, the model predicts a much faster decrease in Ψ_1 with increasing De than the experimental data suggest is correct.

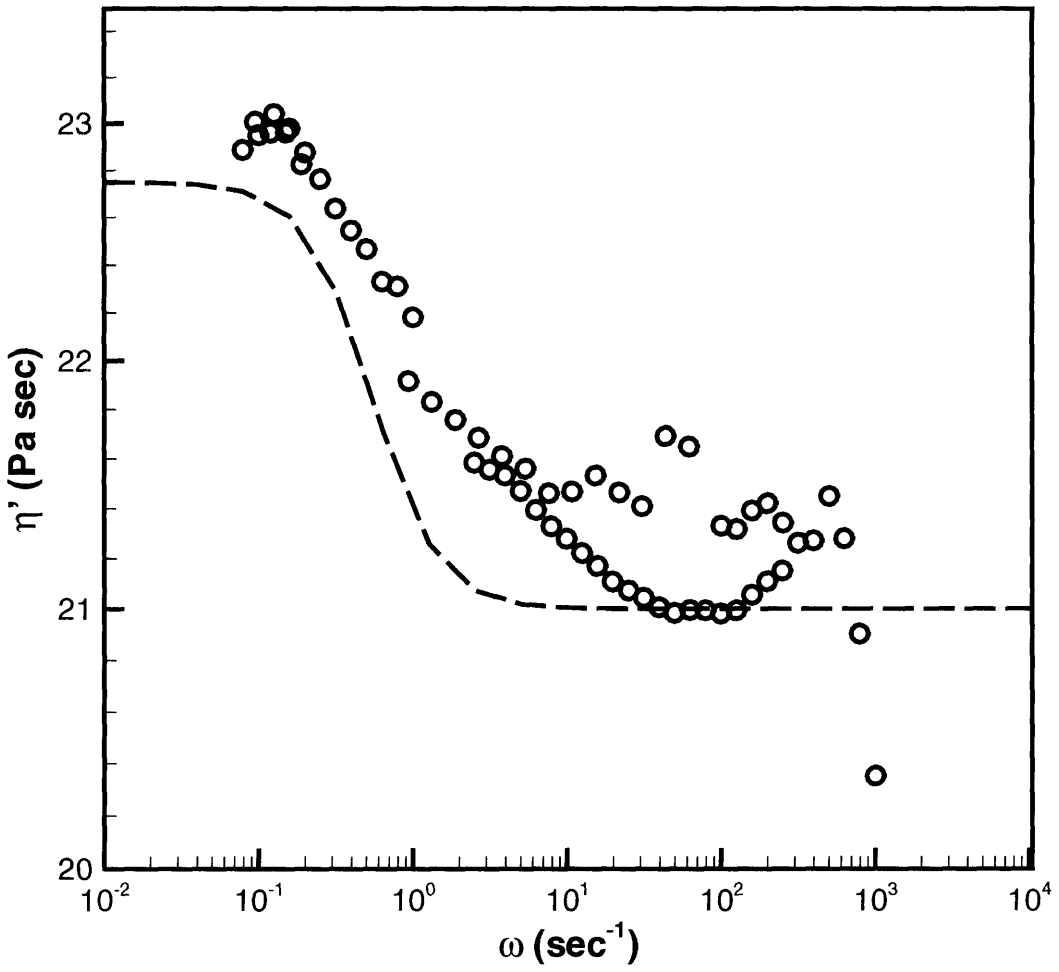


Figure 6-4: Fit of single mode plus solvent to the dynamic viscosity data obtained from experiments. (◯): 0.025% PS/PS fluid; (— —): model.

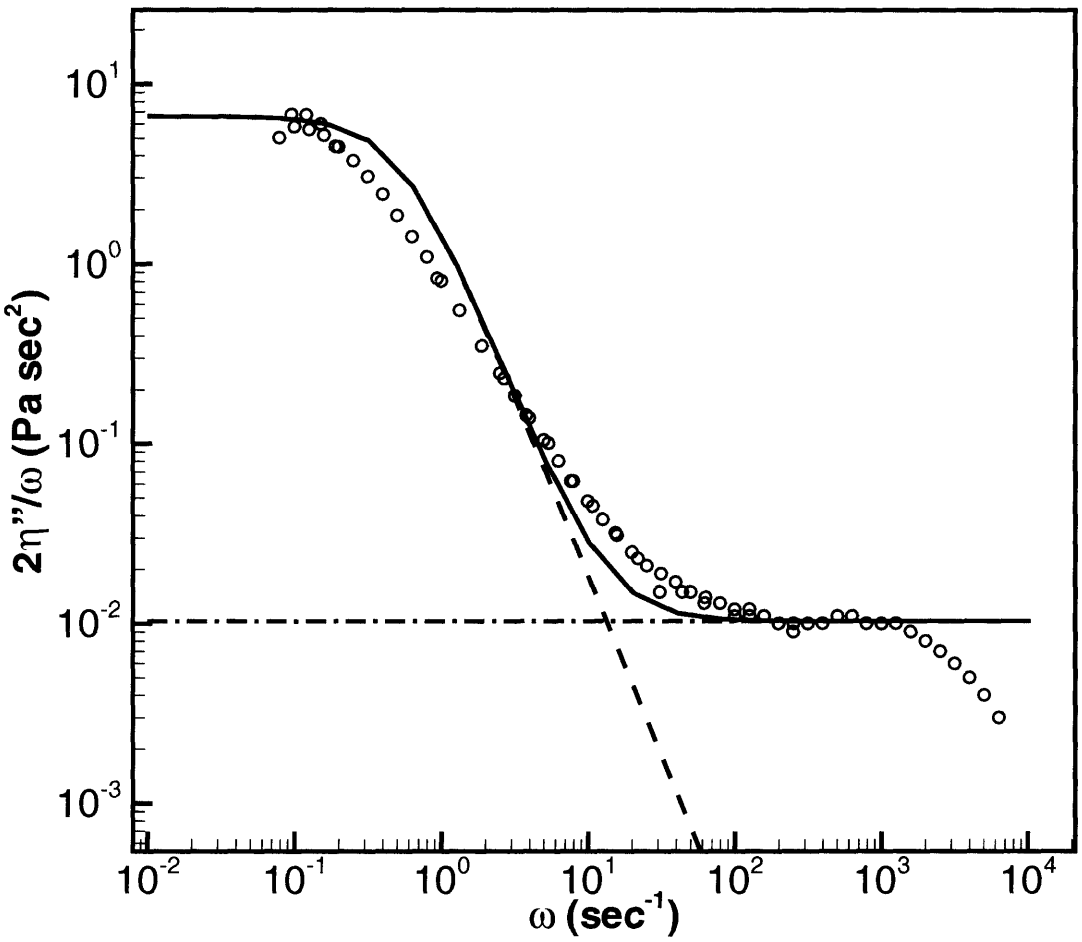


Figure 6-5: Fit of single mode plus solvent to the experimental dynamic rigidity data. From experiments: (○): 0.025% PS/PS fluid. From model: (– –): polymer mode; (– ·): solvent mode; (—): model composite.

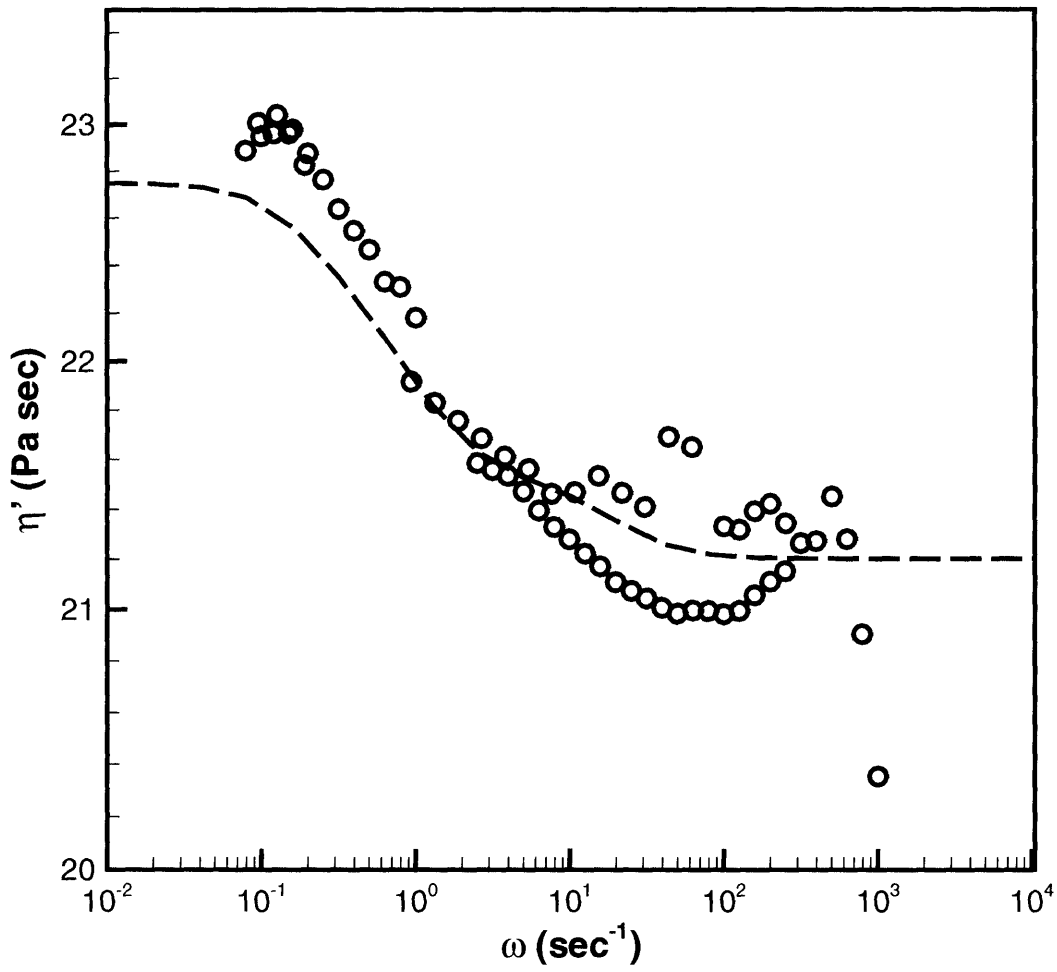


Figure 6-6: Fit of 4 mode plus solvent to the experimental dynamic viscosity data. (◯): 0.025% PS/PS fluid; (— —): model.

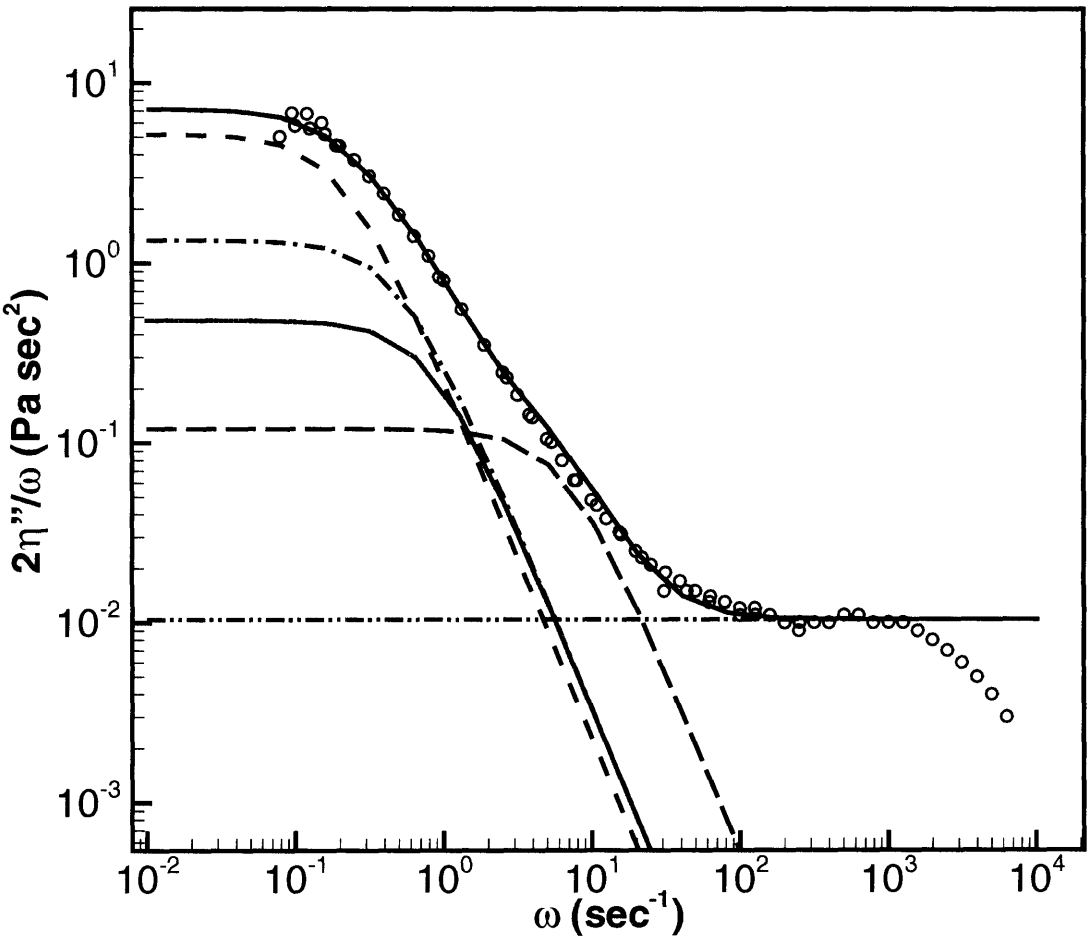


Figure 6-7: Fit of 4 mode FENE-P plus solvent model to the experimental dynamic rigidity data. From experiments: (◯): 0.025% PS/PS fluid. From model: (– –): mode 1; (– ·): mode 2; (·): mode 3; (— —): mode 4; (– ··): solvent mode; (—): model composite.

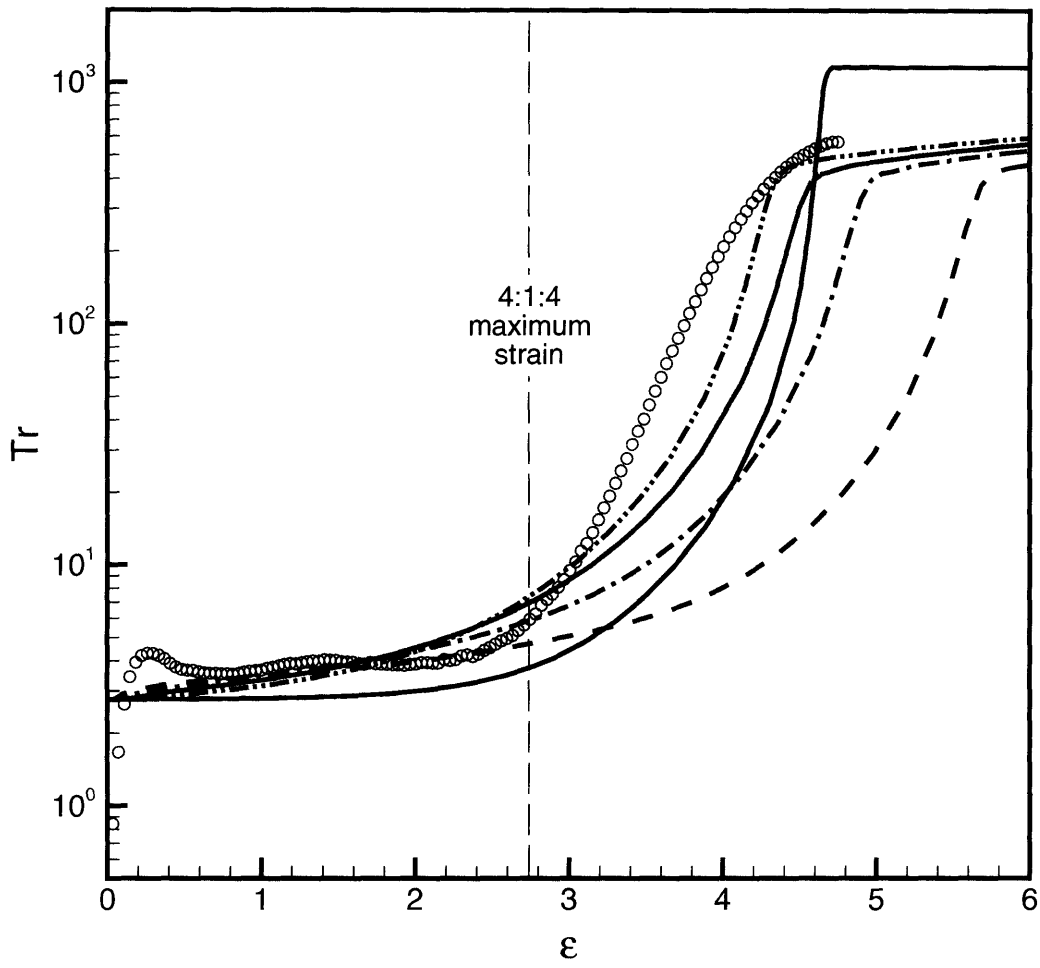


Figure 6-8: Transient extensional viscosity of the single mode models with $b = 7744$. (◯): 0.025% PS/PS fluid; (—): FENE-P; ALS-C model with (– –): $z = 0.25$; (– ·): $z = 0.5$; (·): $z = 1.0$; (– ··): $z = 2.0$.

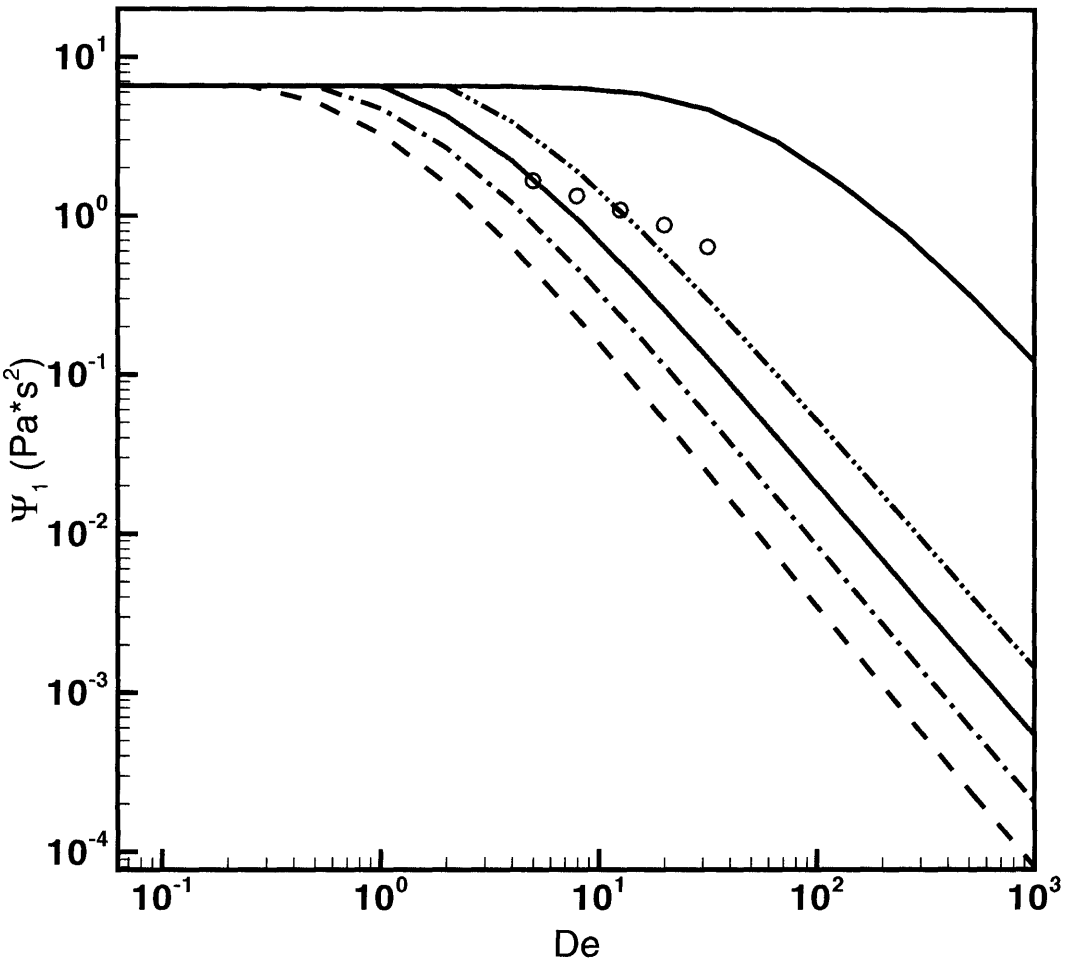


Figure 6-9: First normal stress coefficient for the single mode models with $b = 7744$. (◯): 0.025% PS/PS fluid; (—): FENE-P; (– –): $z = 0.25$; (– ·): $z = 0.5$; (·): $z = 1.0$; (– ··): $z = 2.0$.

The FENE-P model exhibits a similarly quick decrease in Ψ_1 with increasing De, but Ψ_1 for this model begins to decrease at almost a decade higher value of De than that for the ALS-C models.

As is apparent from the comparisons to rheological data, it is difficult to determine the proper value for the parameter z for the ALS-C model. In Figure 6-8 the strain at which the knee occurs for the four choices of z suggests that the correct choice for z is 2.0. However, the maximum strain reached in the 4:1:4 contraction-expansion flow is only 2.75; the ALS-C model with z values of 0.25 and 0.5 better fit the experimental data at strains preceding this value. The plot of Ψ_1 also sheads little light on the appropriate choice for the z parameter. Therefore, a range of values for the parameter z is used in the complex flow calculations in an attempt to better understand the differences in behavior of the model as a function of the z parameter.

For the four-mode model, to obtain a fit to the non-linear rheological data, a small value of b is used for the first mode to represent molecules that are pulled on so quickly by the surrounding flow that they are unable to uncoil and in effect become entangled like a knotted ball of string, whereas b values for the second, third, and fourth mode are chosen at higher values to best fit the transient extensional data. Due to the relatively low maximum strain reached in the 4:1:4 geometry, fit of the transient extensional data only up to a strain of 3 was sought. Fitting the non-linear parameters to the steady-state extensional viscosity was not attempted. The dramatic improvement in the fit to transient extensional viscosity is shown in Figure 6-10. Predictions from the single mode FENE-P and the ALS-C model with $z = 0.5$ are provided for comparison. The plot of the first normal stress coefficient in steady shear for the four mode model is shown in Figure 6-11. Predictions of the single mode models are again provided for comparison. While the four mode model fails to model the experimental data accurately, the trend of the model is much closer than the single mode models.

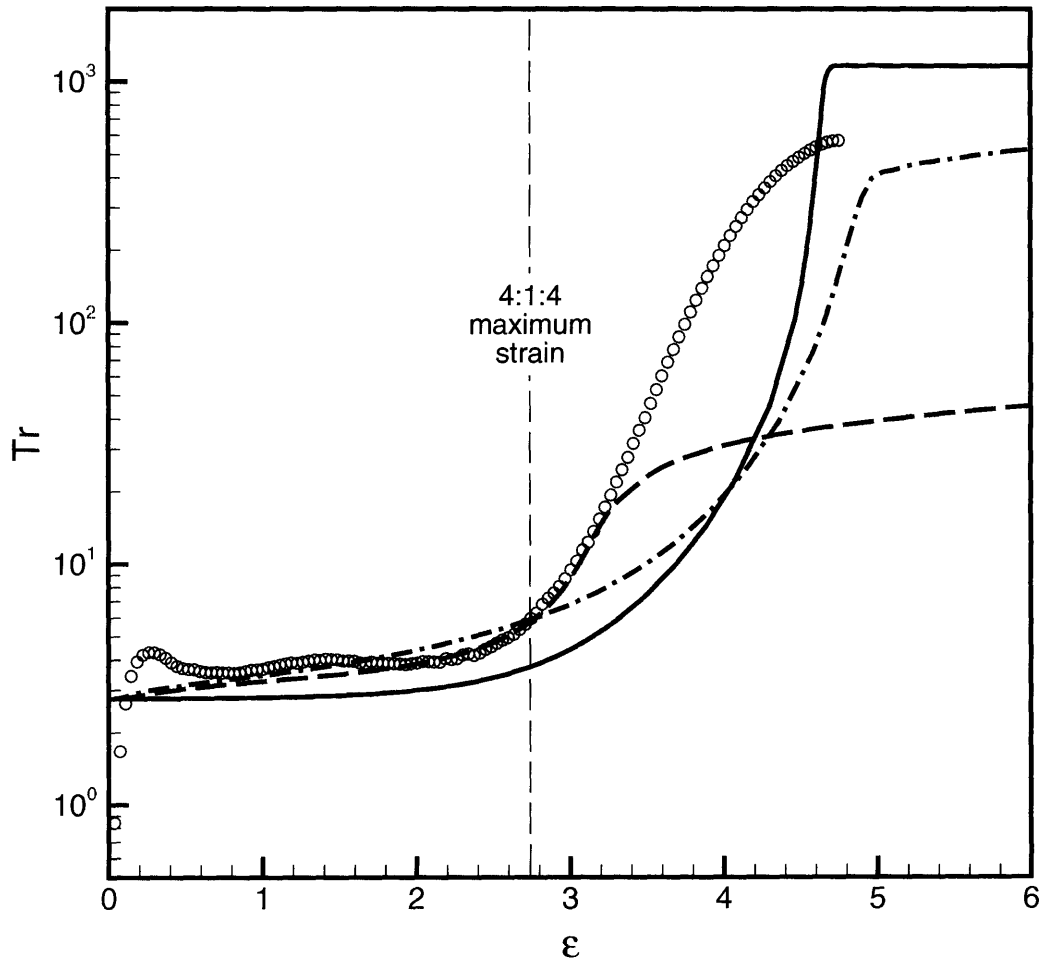


Figure 6-10: Transient extensional viscosity of the 4 mode FENE-P model. (◯): 0.025% PS/PS fluid; (— —): 4 mode FENE-P; (—): 1 mode FENE-P; (– ·): ALS-C with $z = 0.5$.

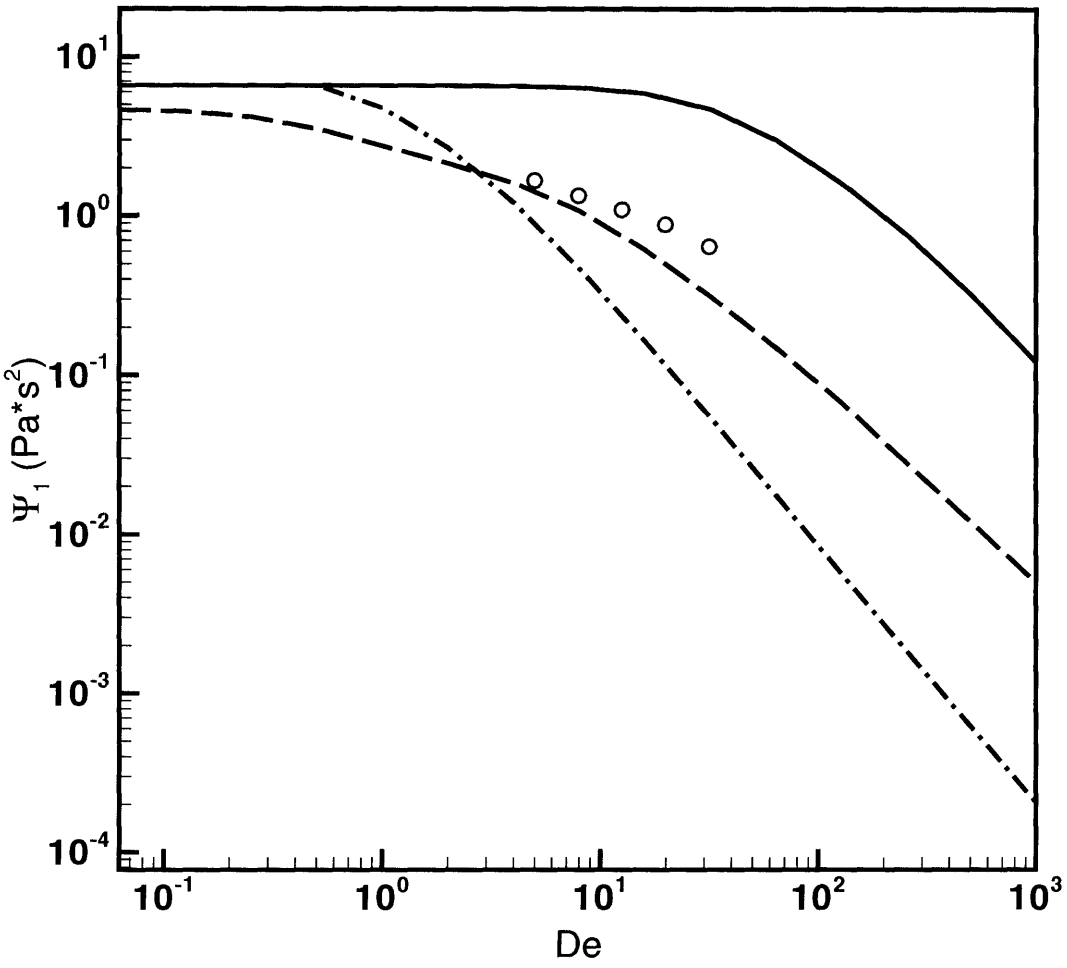


Figure 6-11: First normal stress coefficient for the 4 mode FENE-P model. (◯): 0.025% PS/PS fluid; (— —): 4 mode FENE-P; (—): 1 mode FENE-P; (– ·): ALS-C with $z = 0.5$.

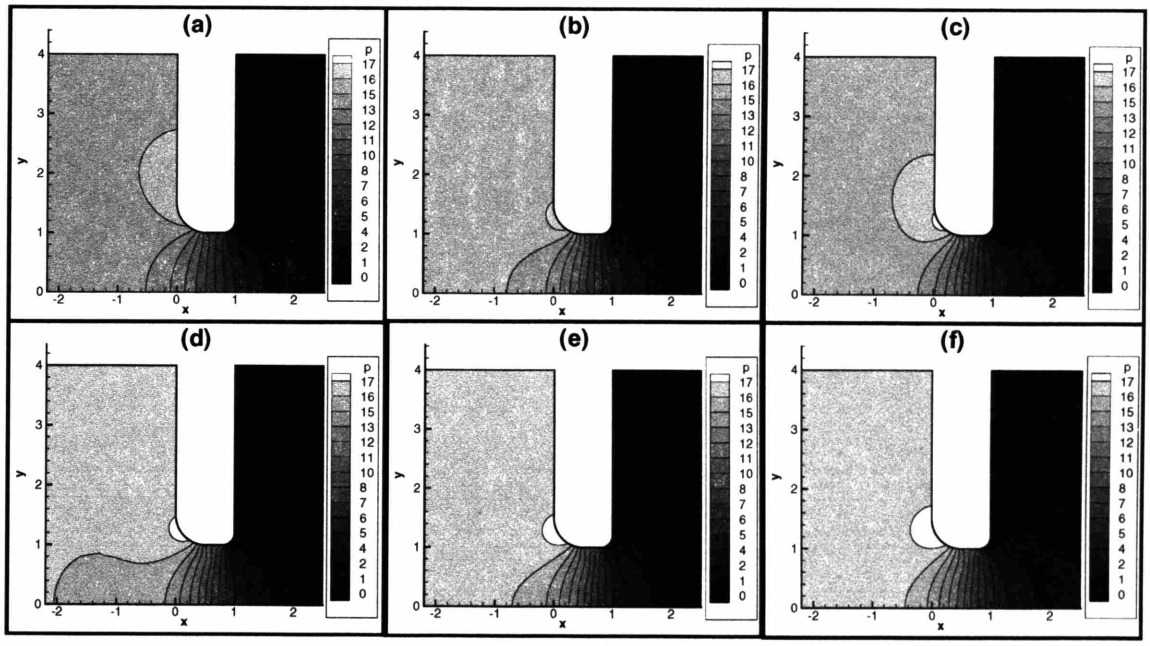


Figure 6-12: Contours of pressure for flow in the 4:1:4 geometry for the ALS-C model with $z = 0.25$. (a): De = 0.5; (b): De = 3.0; (c): De = 6.0; (d): De = 9.0; (e): De = 12.0; (f): De = 16.0.

6.4 Results

6.4.1 Single mode models

Simluation results for the 4:1:4 geometry were generated for the accessible Deborah number space for the ALS-C model with z values of 0.25, 0.5, 1.0, and 2.0 as well as for the FENE-P model. Field plots of the simulation results are presented here for the $z = 0.25$ and $z = 1.0$ results for the ALS-C model. Simulation results for $z = 0.5$ are qualitatively similar to those for $z = 0.25$, and the same is true of the results for $z = 2.0$ as compared to $z = 1.0$. Figure 6-12 shows the pressure contours for a range of De for the $z = 0.25$ results, and Fig. 6-13 shows the corresponding results for $z = 1.0$. As most of the pressure drop in the flow occurs in the contraction throat, an increasing fraction of the total pressure drop is seen near the rounded reentrant corner as De is increased for both

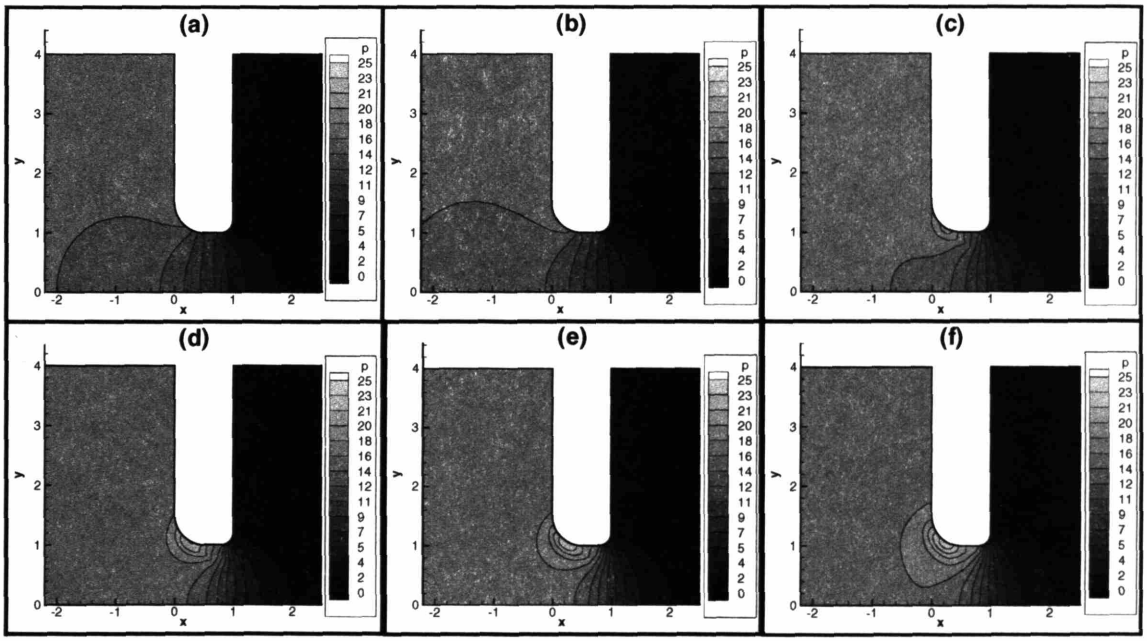


Figure 6-13: Contours of pressure for flow in the 4:1:4 geometry for the ALS-C model with $z = 1.0$. (a): De = 0.5; (b): De = 3.0; (c): De = 6.0; (d): De = 9.0; (e): De = 12.0; (f): De = 16.0.

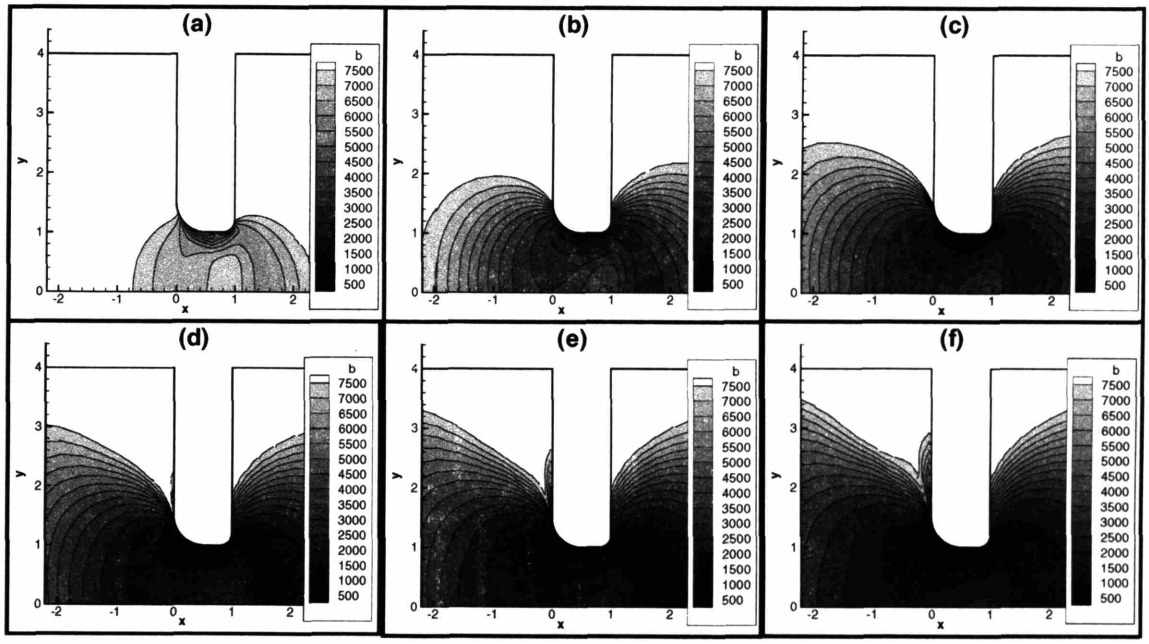


Figure 6-14: Contours of $\langle b_{seg} \rangle$ for flow in the 4:1:4 geometry for the ALS-C model with $z = 0.25$. (a): De = 0.5; (b): De = 3.0; (c): De = 6.0; (d): De = 9.0; (e): De = 12.0; (f): De = 16.0.

the $z = 0.25$ and $z = 1.0$ cases. However, in the $z = 1.0$ case the maximum pressure occurs within the small tube, while in the $z = 0.25$ case the maximum pressure occurs farther upstream at the entrance to the small tube. Figure 6-14 shows the segment length contours for a range of De for the $z = 0.25$ results, and Fig. 6-15 shows the corresponding results for $z = 1.0$. The majority of the decrease in segment length, or conversely the increase in the average number of segments per molecule, occurs near the wall of the contraction throat. This is the area of maximum shear rate in the flow. Travelling along the centerline of the geometry, the average molecule moves from a region of few segments upstream of the contraction, through a region of increased segments near the contraction entrance at $x = 0$, next to a region of fewer segments again near the middle of the contracted section, then to a region of increased segments again just beyond the exit plane of the contraction, and finally to a region where the number of segments re-

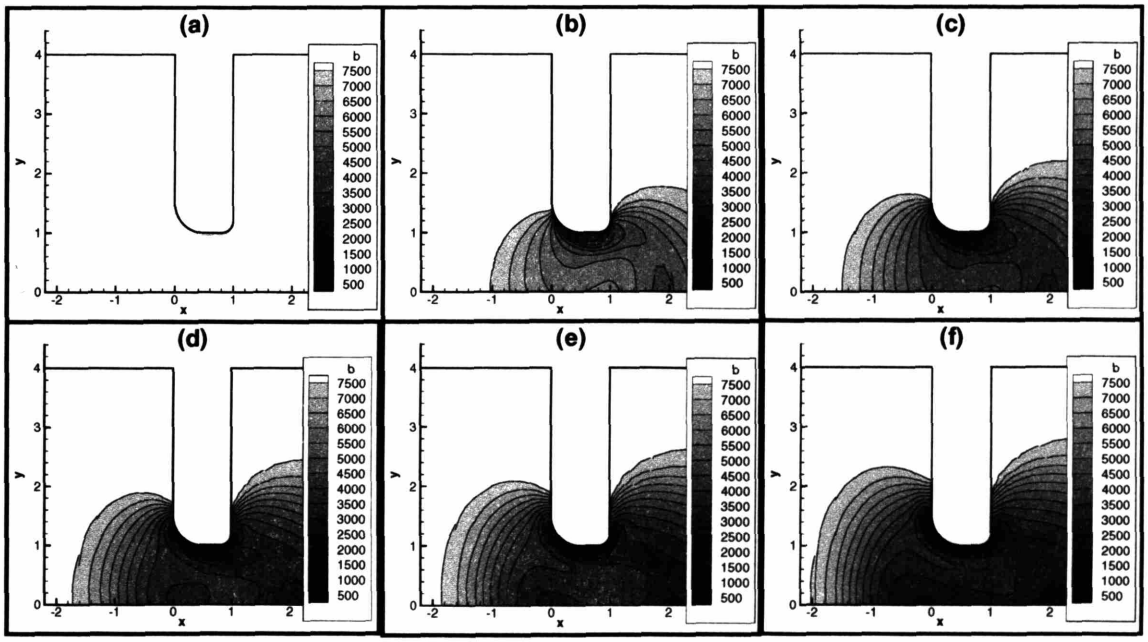


Figure 6-15: Contours of $\langle b_{seg} \rangle$ for flow in the 4:1:4 geometry for the ALS-C model with $z = 1.0$. (a): De = 0.5; (b): De = 3.0; (c): De = 6.0; (d): De = 9.0; (e): De = 12.0; (f): De = 16.0.

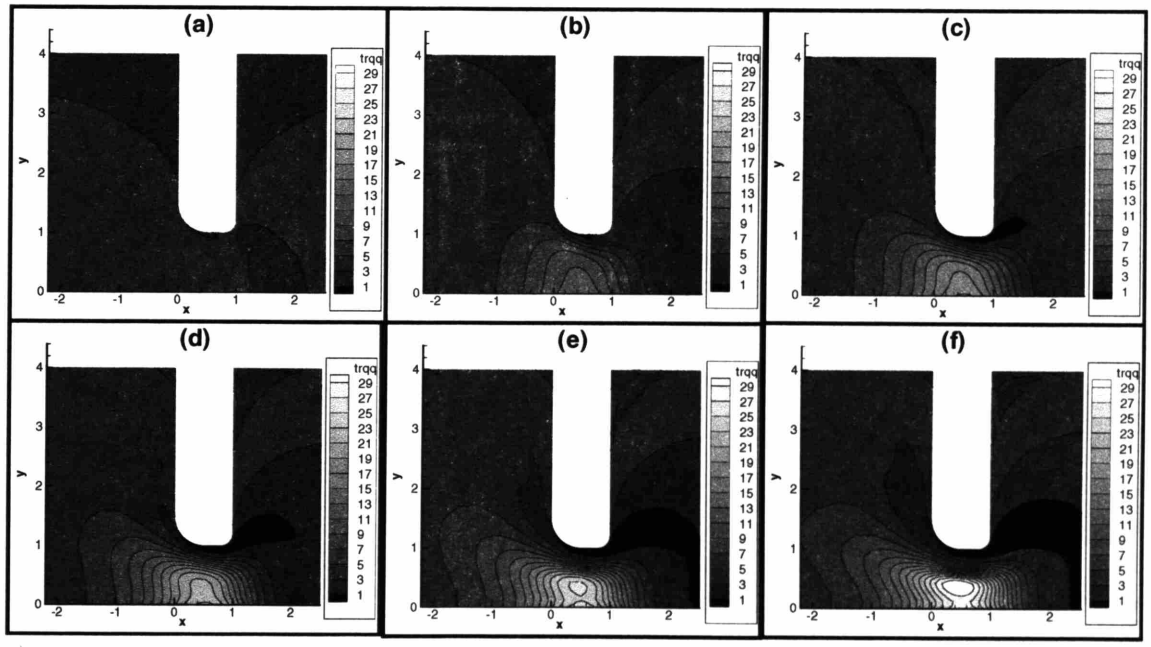


Figure 6-16: Contours of $tr\langle \mathbf{QQ} \rangle$ for flow in the 4:1:4 geometry for the ALS-C model with $z = 0.25$. (a): De = 0.5; (b): De = 3.0; (c): De = 6.0; (d): De = 9.0; (e): De = 12.0; (f): De = 16.0.

turns to a value near 1 in the region downstream of the contraction. This trend is true for both the $z = 0.25$ and $z = 1.0$ cases, and the changes become more dramatic as De is increased. To gain a better understanding of this behavior, analysis of the data for $\langle b_{seg} \rangle$ can be coupled with data of the trace of the average configuration of the molecules, $tr\langle \mathbf{QQ} \rangle = \langle \mathbf{QQ} \rangle_{xx} + \langle \mathbf{QQ} \rangle_{yy} + \langle \mathbf{QQ} \rangle_{zz}$. Figure 6-16 shows the contours of $tr\langle \mathbf{QQ} \rangle$ for a range of De for the $z = 0.25$ results, and Fig. 6-17 shows the corresponding results for $z = 1.0$. By looking also at the trace of the configuration tensor in the flow, it is clear that, on average, molecules are being stretched out from a coiled state to a highly elongated state as they pass through the contraction on the centerline. The behavior of the number of segments shown in figures 6-14 and 6-15 is expected, since the ALS-C model uses a single segment to represent both the coiled state and the highly elongated state of the molecule. Hence, whereas the number of segments passes through three

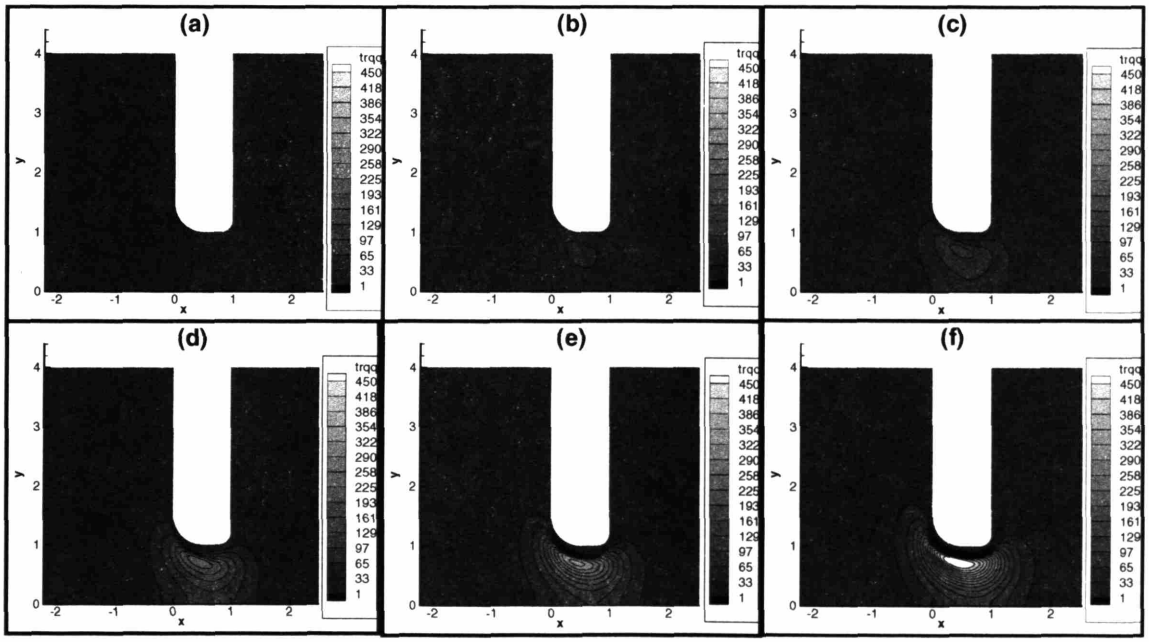


Figure 6-17: Contours of $tr\,\langle \mathbf{QQ} \rangle$ for flow in the 4:1:4 geometry for the ALS-C model with $z = 1.0$. (a): De = 0.5; (b): De = 3.0; (c): De = 6.0; (d): De = 9.0; (e): De = 12.0; (f): De = 16.0.

different extrema as the flow progresses through the contraction near the centerline, the average length of the molecules passes through only one extremum. The major difference between the simulation results of the $z = 0.25$ and $z = 1.0$ cases for the ALS-C model is the amount of molecular stretch exhibited in the contraction throat. The results for the $z = 1.0$ case show very high values of $tr\,\langle \mathbf{QQ} \rangle$ in the region between the contraction wall and centerline, whereas the $z = 0.25$ case shows moderate values of $tr\,\langle \mathbf{QQ} \rangle$ concentrated around the centerline in the contraction. The high values of $tr\,\langle \mathbf{QQ} \rangle$ in the $z = 1.0$ case also seem to affect the flow around the contraction region, as seen in the plots of the pressure field for the $z = 1.0$ case, Fig. 6-13. A significant amount of pressure buildup can be seen near the reentrant corner leading into the region of high $tr\,\langle \mathbf{QQ} \rangle$, as though extra resistance is being added to the flow in the region of high $tr\,\langle \mathbf{QQ} \rangle$, manifested by a buildup of pressure just upstream of this region.

The macroscopic property of the flow that is of interest in comparison to the experimental findings is the extra pressure drop generated by the fluid entering and exiting the contracted region of the geometry. This quantity can be extracted from the overall pressure drop in the geometry by subtracting out the pressure drop due to the Poiseuille flow in the upstream, downstream, and contraction sections of the geometry. The contracted region here is assumed to be a straight section of pipe, an assumption made to agree with the work of Rothstein et al. [65]. The resulting expression for the extra pressure drop is given in equation 6.14.

$$\Delta P'_{ud} = (P_d - P_u) - \frac{8Q\eta}{\pi}\left(\frac{L}{R_1^4} + \frac{L_c}{R_2^4}\right) \tag{6.14}$$

where $L \equiv (z_u - z_d) - L_c$ is the total length of pipe with radius R_1 in the geometry. Again in agreement with Rothstein et al. [65], the extra pressure drop is scaled by the extra pressure drop of a Newtonian fluid flowing through a 4:1:4 contraction-expansion with sharp corners. The extra pressure drop of the Newtonian fluid is found via simulation of this geometry.

Figure 6-18 shows the computed extra pressure drop through the channel as a function

of Deborah number as compared to the experimental findings presented in [65]. Rothstein et al. found that as De was increased the extra pressure drop first decreased slightly then increased monotonically until the flow became unstable around a De of 40 for the flow in the 4:1:4 contraction-expansion geometry with rounded corners. For the limited De range over which solutions can be obtained for the FENE-P model, a decrease in pressure drop is found, in contrast to the experimental findings. For all choices of z parameter for the ALS-C model, pressure drop enhancement with increasing De is observed in qualitative agreement with the experimental findings. However, none of the choices of z yield quantitative agreement with the experimental findings.

A qualitative difference is seen in the pressure drop with increasing De for the range of z's used. The pressure drop for $z = 1.0$ shows two different distinct increasing regions with inflections near De of 8 and 14 corresponding with the appearance and dissappearance of a ridge of pressure in the contraction throat (most prevelant in Fig. 6-13(d)), whereas results for z of 0.25 and 0.5 show only one increasing region followed by a plateau and no appearance of a pressure ridge. The increasing pressure drop of the model with $z = 2.0$ shows behavior similar, though muted, to the $z = 1.0$ model.

In order to compare the simulation results to the experimentally measured upstream vortex growth, the streamlines of the flows are computed over the range of Deborah numbers simulated. Rather than computing the stream function for the given flow, data of the components of velocity are taken from the simulation and combined to form a vector field over the flow domain. Streamtraces are then found much like experimental streakline images are constructed, by tracing the path of marker particles flowing through the domain. An example of a streamtrace plot for the 4:1:4 geometry is shown in Fig. 6-19. Since the solutions presented here have been time integrated to steady state, the streamtraces found in this manner are equivalent to the streamlines of the flow. It is important to note that while streamlines found in this manner are valid for the given flow, changes in the concentration of streamlines here do not indicate changes in the intensity of the flow, as would be the case with streamlines plotted from evenly spaced

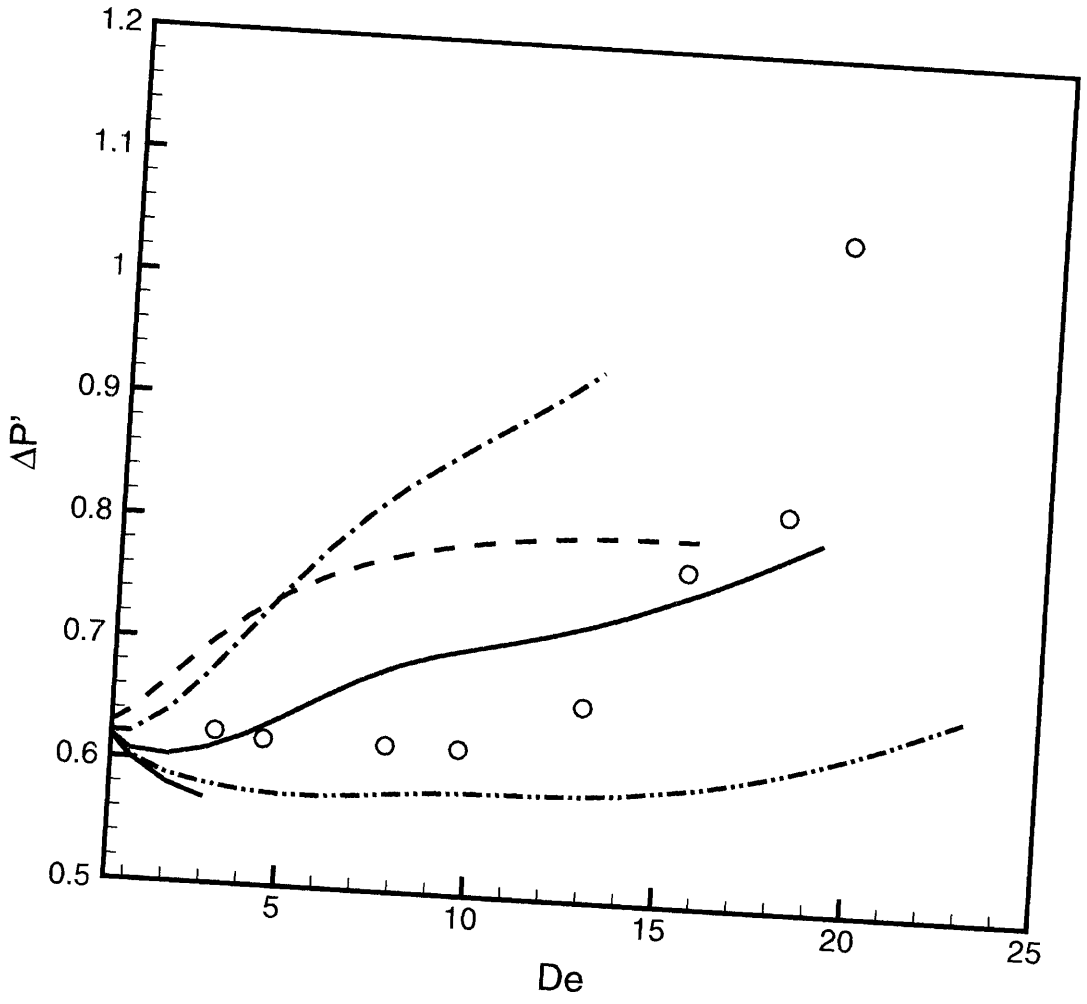


Figure 6-18: Extra pressure drop of the single mode fluid the 4:1:4 contraction-expansion geometry. (◯): 0.025% PS/PS fluid; (—): FENE-P; (– –): $z = 0.25$; (– ·): $z = 0.5$; (·): $z = 1.0$; (– · ·): $z = 2.0$.

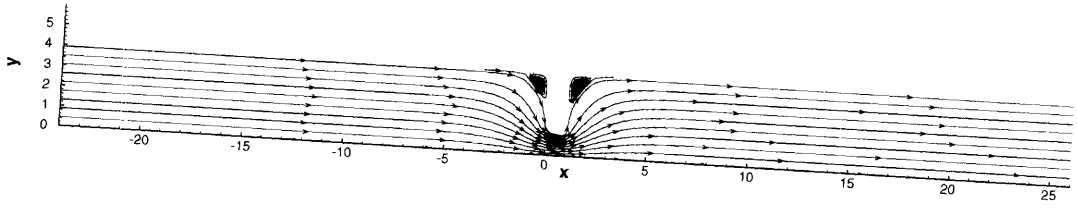


Figure 6-19: Sample streamtraces for flow in the 4:1:4 geometry.

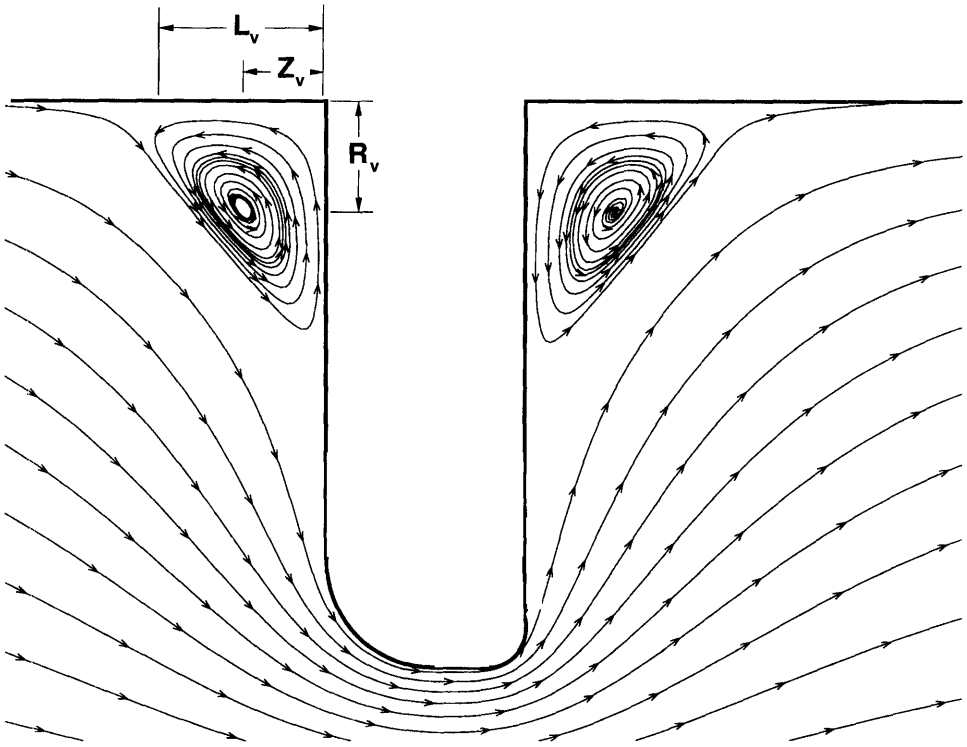


Figure 6-20: Key measurements for characteristics of upstream corner vortex. Z_v and R_v are the axial and radial distance, respectively, of the vortex center from the salient corner upstream of the contraction. L_v is the vortex reattachment length.

values of the streamfunction. The pertinent information regarding the formation and growth of the vortex in the salient corner is shown schematically in Fig. 6-20, namely the location of the upstream vortex center relative to the salient corner and the vortex reattachment length.

Figure 6-21 illustrates the vortex growth over a range of De for $z = 1.0$. The flow pattern in the salient corner progresses from a Moffat vortex at low De shown in subfigure (a) to a corner vortex shown in subfigure (b). The flow then slowly recedes back into the salient corner as De is further increased. Just prior to failure of the calculations signs of the formation of a lip vortex near the reentrant corner were seen in the calculations. A

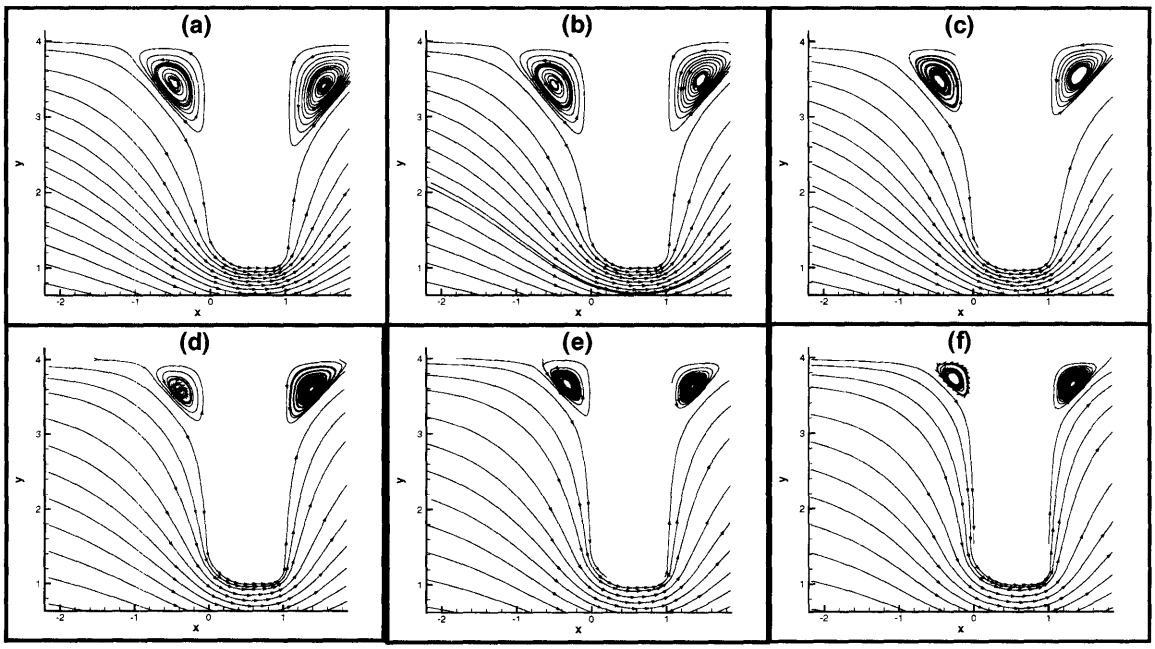


Figure 6-21: Streamlines near the salient corner of the 4:1:4 contraction-expansion with increasing Deborah number for the ALS-C model with $z = 1.0$. (a): $De = 0.5$; (b): $De = 3.0$; (c): $De = 6.0$; (d): $De = 9.0$; (e): $De = 12.0$; (f): $De = 16.0$.

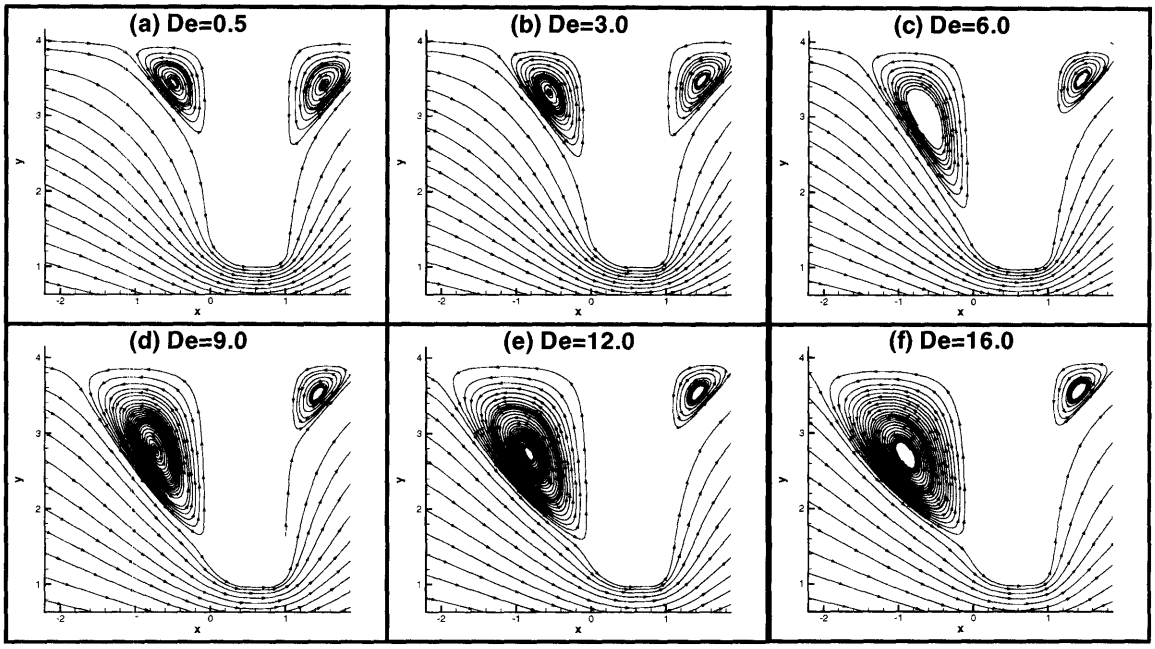


Figure 6-22: Streamlines near the salient corner of the 4:1:4 contraction-expansion with increasing Deborah number for the ALS-C model with $z = 0.25$. (a): $De = 0.5$; (b): $De = 3.0$; (c): $De = 6.0$; (d): $De = 9.0$; (e): $De = 12.0$; (f): $De = 16.0$.

lip vortex has never boon found, however, in a steady-state solution in the calculations presented herein.

Figure 6-22 illustrates the vortex growth over a range of De for $z = 0.25$. The flow pattern in the upstream corner progresses from a Moffat vortex with concave streamlines in (a), to a corner vortex with convex streamlines in (b). Steady growth is then seen in (c) through (f). This behavior agrees qualitatively with the 4:1:4 experiments in [65]. However, as with the trends of computed extra pressure drop, these choices for z are unable to produce quantitatively accurate results when compared to the experimental findings concerning the growth of the salient corner vortex.

Trends similar to those of the extra pressure drop are seen in the uptream vortex behavior with increasing De of the ALS-C model simlulations for the four choices of z. For $z \geq 1$, the upstream vortex is seen to shrink back into the corner as De is

increased. Figure 6-23 displays the key characteristics of the vortex formed in the upstream salient corner for the ALS-C fluid. Two distinct decreasing regions can be seen in the reattachment length of the corner vortex in the same range of De where an increase is seen in the pressure drop. However, these trends are in qualitative disagreement with the experimental findings and suggest the formation of a lip vortex is imminent. Though this was not observed in the 4:1:4 experiments, it seems relevant to note that it was observed experimentally in the 2:1:2 contraction-expansion case [65]. The cases of $z < 1$, however, show quite different behavior for the upstream corner vortex. In both of these cases, the vortex is seen to move radially and axially away from the corner and increase in size. Figure 6-24 shows the characteristics of the vortex in the salient corner of the 4:1:4 geometry as reported in [65]. From comparison to the experimental data, it is clear that although the $z < 1$ cases show the correct trend for the salient corner vortex, namely growth and movement away from the salient corner, the models still fail to predict quantitatively the dynamics of the salient corner vortex.

6.4.2 4-mode FENE-P model

Simulation results using a 4-mode FENE-P model for the flow in the 4:1:4 contraction-expansion geometry are presented here. Figure 6-25 shows the computed extra pressure drop through the channel as a function of Deborah number as compared to the experimental findings presented in [65]. Convergent solutions for the 4-mode model were attained for much higher De than with the single mode FENE-P model. This is a direct result of the smaller b values used for the finite extensibility of each mode of the model as compared to the single mode finite extensibility of $b = 7,744$ chosen to fit the steady-state extensional viscosity. Over the range of De where solutions were computed, the simulations exhibit an initial decrease in the extra pressure drop followed by a monotonic increase. Eventually a plateau region is reached at higher De. While the trend is similar to that of the single mode ALS-C model, especially the $z = 0.25$ case, the results of the 4-mode FENE-P model show much less increase in extra pressure drop, and they reach

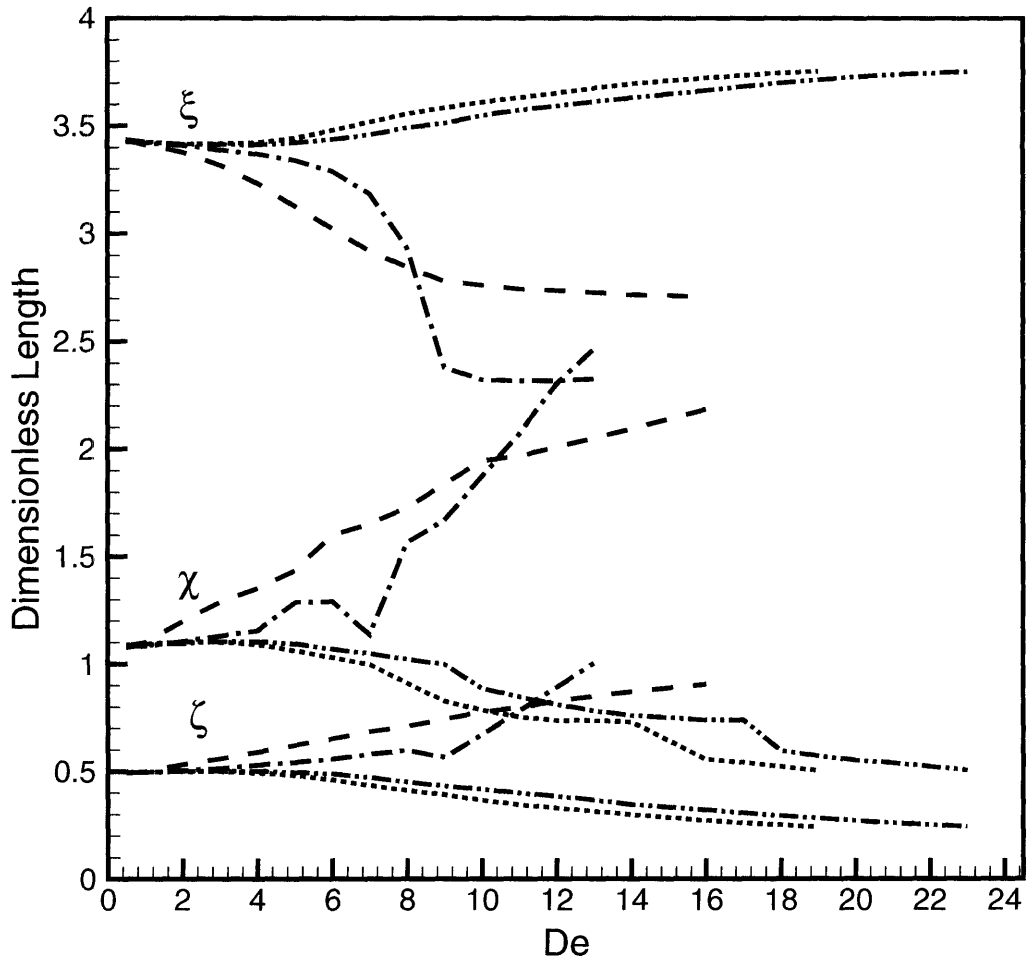


Figure 6-23: Key characteristics of the vortex formed in the upstream salient corner with increasing Deborah number for the ALS-C fluid where $\chi = L_v/R_2$ is the vortex reattachment length, $\xi = R_v/R_2$ is the radial location of vortex center, and $\zeta = Z_v/R_2$ is the upstream location of the vortex center. (– –): z=0.25; (– ·): z=0.5; (·): z=1.0; (– · ·): z=2.0.

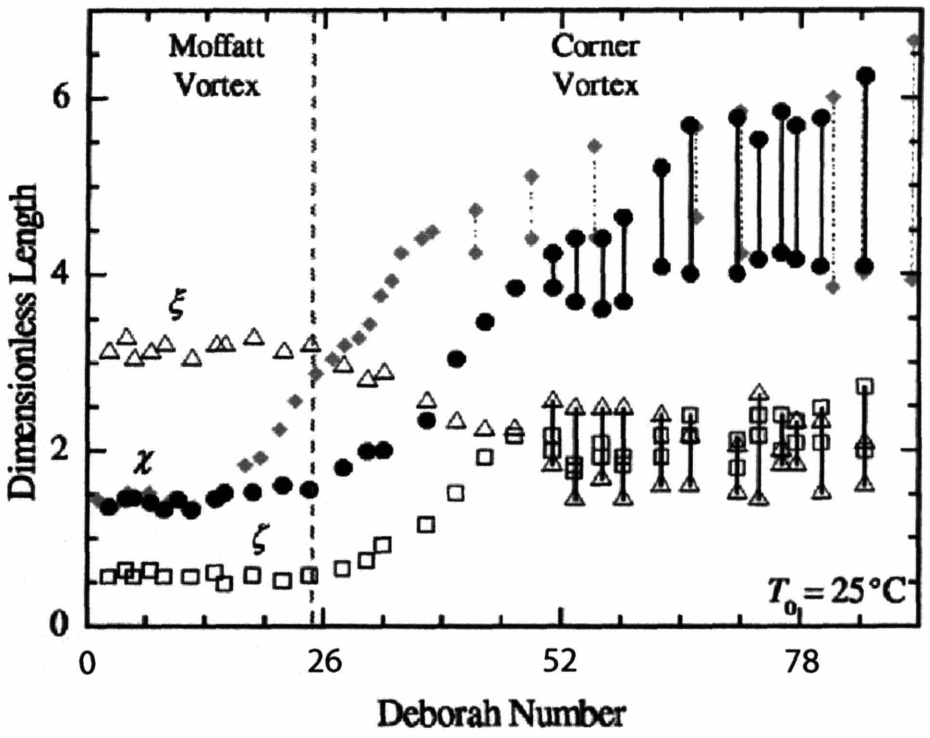


Figure 6-24: Characteristics of the upstream growth dynamics as a function of Deborah number for the 4:1:4 axisymmetric contraction-expansion with rounded entrance lip, $R_c = 0.5R_2$ - (•): vortex reattachment length, $\chi = L_v/R_2$; (△): radial location of vortex center, $\xi = R_v/R_2$; (□) the upstream location of the vortex center, $\zeta = Z_v/R_2$; and (♦): vortex reattachment length for the 4:1:4 contraction-expansion with sharp entrance lip, $R_c = 0$. Reproduced from [65]. Note: De scale has been modified to correspond with the current work.

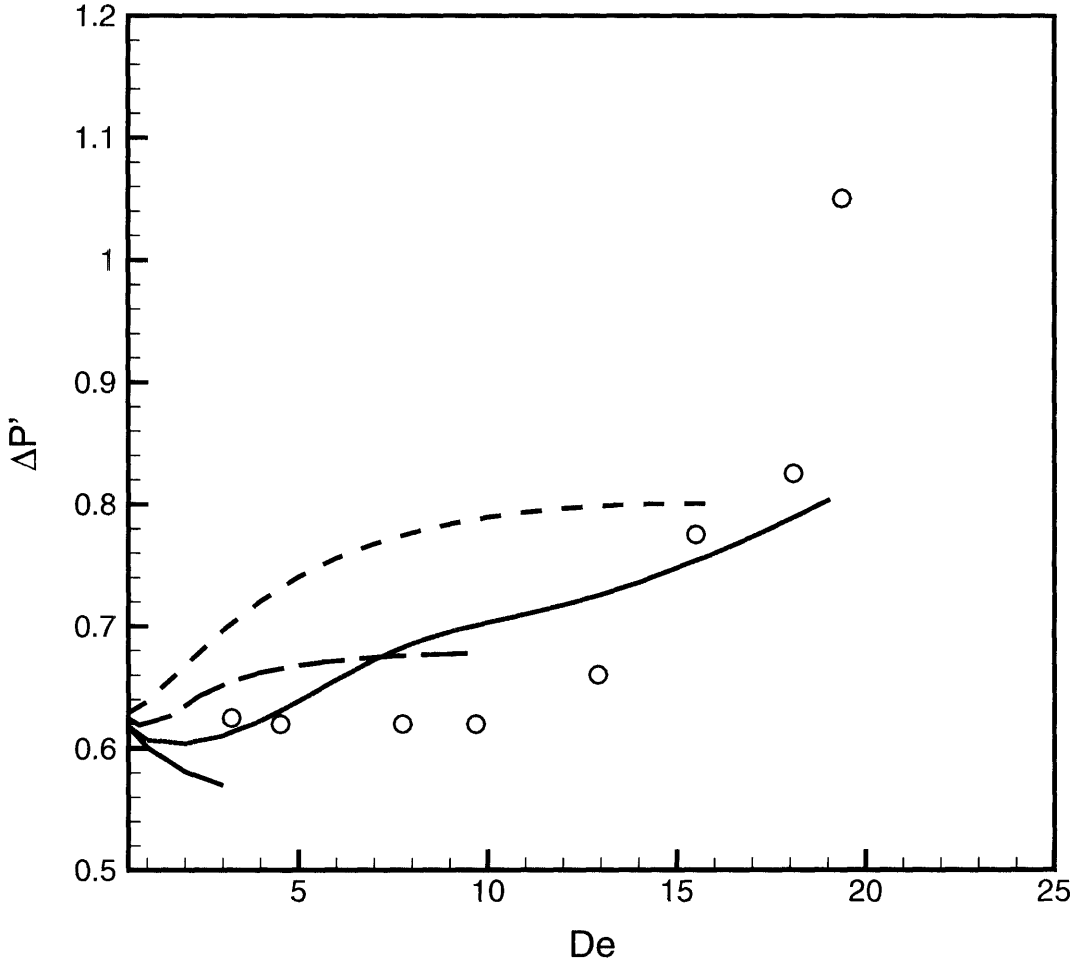


Figure 6-25: Extra pressure drop of the single and 4-mode models in the 4:1:4 contraction-expansion geometry. (◯): 0.025% PS/PS fluid; (— —): 4-mode FENE-P; (—): FENE-P; ALS-C with (– –): $z = 0.25$; (·): $z = 1.0$.

the plateau region at a De of about half of that in the ALS-C $z = 0.25$ case.

Figure 6-26 displays the key characteristics of the vortex formed in the salient corner for the simulations using the 4-mode FENE-P model. Similar to the behavior of extra pressure drop with increasing De, the location of the vortex center and the reattachment length in the 4-mode FENE-P simulations show the same trend as the $z < 1$ ALS-C calculations, namely, the vortex center moves out and away from the salient corner and the reattachment length steadily increases with increasing De. However, the behavior of the salient corner vortex with increasing De is significantly muted as compared to the experimental findings as well as the ALS-C calculations presented here.

6.5 Conclusions

For the first time a model that captures the increase in pressure drop with increasing flow rate is presented for the flow of Boger fluids in axisymmetric contraction-expansion geometries. Using the ALS-C model, simulations were carried out for a range of the order one z parameter. Pressure drop enhancement was predicted for all values of z. Analysis of the dynamics of the salient corner vortex with increasing Deborah number in the cases with $z \geq 1$ show that the vortex receeds into the salient corner as De is increased. By contrast the $z < 1$ cases show a vortex that grows and moves radially and axially away from the salient corner. Comparison to experimental observations of Rothstein et al. [65] shows that the $z < 1$ cases yield the qualitatively correct trend for the dynamics of the salient corner vortex.

Although the ALS-C model with $z < 1$ correctly captures the qualitative behavior of the salient corner vortex and predicts pressure drop enhancement with increasing De, the model does not quantitatively describe the behavior. In an attempt to describe better the behavior of the fluid, a 4-mode FENE-P model is used. Comparison to the rheological data show vast improvement in the ability of the multimode model to describe the Rothstein fluid in simple homogeneous flows. Like simulations for the ALS-

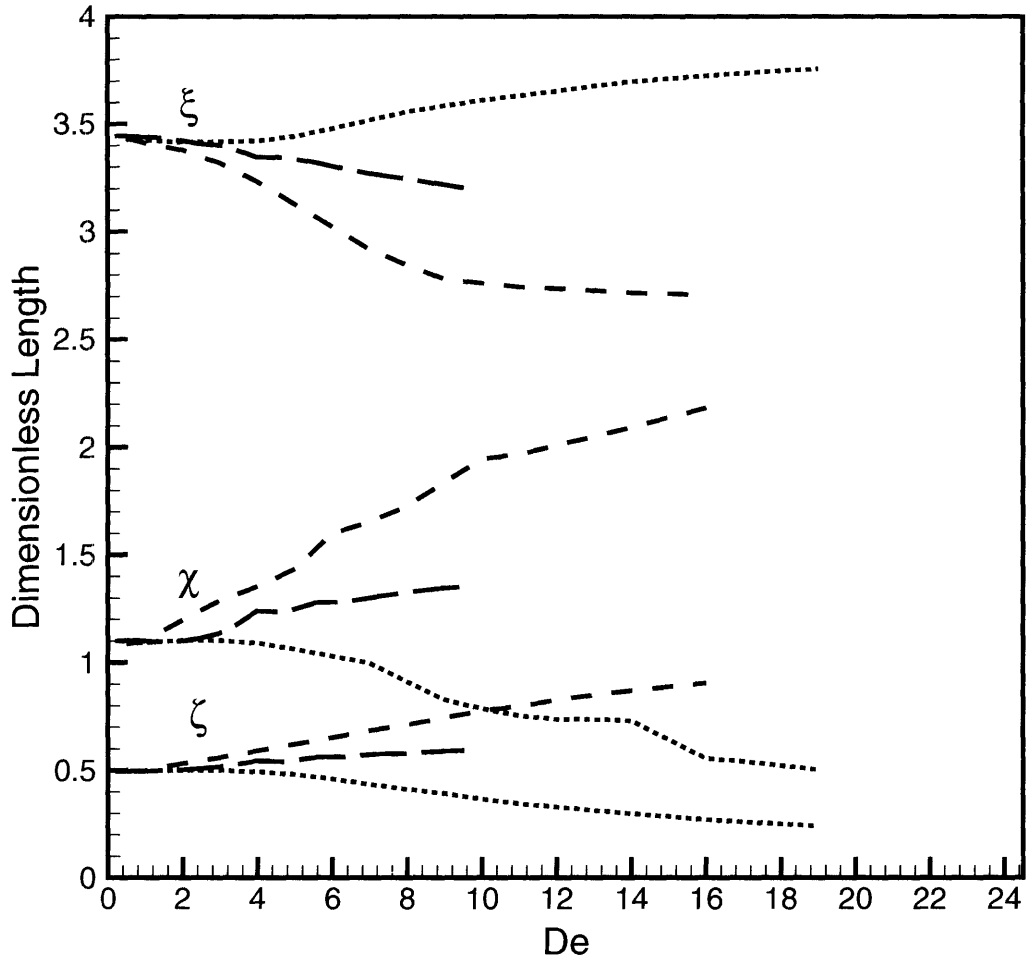


Figure 6-26: Key characteristics of the vortex formed in the salient corner with increasing Deborah number for the single and 4-mode fluids. 4-mode FENE-P: (— —); ALS-C: (– –): z=0.25; (.): z=1.0.

C model, simulations using the multimode model yeild the correct trend for both the extra pressure drop with increasing De as well as predict growth of the salient corner vortex with increasing De. However, the multimode model also fails to predict quantitatively either the pressure drop or the vortex growth observed experimentally.

Chapter 7

Three-Dimensional Finite Element Method for Confined Flows

7.1 Motivation

Development of three-dimensional structures from the growth of instabilities in the flow of viscoelastic fluids is common throughout the literature. As shown by Evans and Walters [31], [32] entry flows into a planar contraction of width to height ratio of 1 to 2 yields full three-dimensional structures, a phenomena that could not possibly be captured with a two-dimensional simulation. In a series of papers, Chiba et al. ([18], [19], [17]) examined the formation and motion of Goertler vortices, longitudinal vortex tubes forming near the wall of the contraction plane, for the entry flow of dilute and semidilute polyacrylamide solutions in a planar contraction geometry. They found that the presence of three-dimensional lip vortices play an important role in the formation of the Goertler vortex pairs [17]. Gesenier [33] also explored the effects of contraction ratio on instabilities in the planar contraction. While the mechanism of the instabilty is not fully understood, it is apparent that the boundedness in the "neutral" direction is not likely the cause given that the instability exists for a wide range of width to height ratios. Another such instability that has been well studied is the flow around period array of

cylinders of various spacings [3], [54], [57], [71]. For the infinitely spaced case, that of an isolated cylinder in a channel, McKinley et al. [57] observed through experimentation three dimensional spatially periodic structures on the downstream side of the cylinder.

Aside from the need for three-dimensional simulators to capture flow structures that arise from instabilities, there are also many industrially important problems in which the three-dimensional structure of the problem arises from the problem geometry alone. An example of this is fiber spinning. Traditional fiber spinning consists of extrusion and draw down of a cylindrical filament from a die. By assuming the filament and die are both axisymmetric, the model is reduced from three dimensions to two dimensions [70]. However, this is rarely the case in the fiber spinning industry. A variety of cross sections are often used to enhance properties such as touch and feel.

Even in the case of circular cross section fibers, because of the high processing rates used in industry, it is common practice to draw multiple fibers from a single die assembly and then draw the fibers together into a bundle. Die plates with three, five, or more die holes are used for this application. While each die hole would behave similarly to an axisymmetric contraction flow, developing lip vortices [8], it is unclear what would happen as the lip vortices from two adjacent die holes collide with one another, but certainly potentially rich behavior such as this cannot be captured by simulation of the simple one hole die. In addition, there are important issues of distribution of polymer among the die holes, which can only be captured with a three-dimensional simulation.

It is clear that the need for three-dimensional modeling of the dynamics of viscoelastic fluids is important for the understanding and development of many applications. Work to this point in the field has included mostly finite volume simulations on coarse meshes [60], [87] with some attempts in finite elements on even coarser meshes [8].

The purpose of this chapter is to detail the implementation of a time-dependent, three-dimensional finite element package for the solution of viscoelastic flows in confined geometries. Section 7.2 details the governing equations and boundary conditions used. Section 7.3 describes the finite element method and associated basis functions

and quadrature used to discretize the governing equations. Section 7.4 describes results from comparisons of the method to analytical solutions and other proven methods. Section 7.5 defines the parallel method used and gives measures of the performance of the method.

7.2 Problem Description

7.2.1 Governing Equations

Discussion of the equations governing the flow of viscoelastic fluids can be found in Section 2.3. The method uses the DEVSS-G formulation discussed in Section 3.1 with the decoupled G formulation discussed in Chapter 4. While the method is designed to accept a wide variety of constitutive equations, the Giesekus equation 2.26a is chosen for comparison and performance measurements presented in this chapter.

7.2.2 Boundary Conditions

Given that the system of equations describing the flow of a viscoelastic fluid consists of partial differential equations, it is necessary to prescribe boundary conditions on the equations where applicable. Two boundary conditions are necessary for each component of velocity, given that it is described by a second order partial differential equation. Typically only one boundary condition is necessary for the components of stress, given that they are usually described by a first order partial differential equation. For the constitutive equations used within this work, this is always the case.

Allowing for a wide range of possible boundary conditions helps to make a fairly robust simulation package capable of modeling many different geometries. For confined flows, the implemented boundary conditions include no-slip surfaces, reflective symmetry surfaces, periodic surfaces, two-dimensional inflow surfaces and fully-developed outflow surfaces. Details of conditions specified at each boundary follow.

The no-slip boundary condition consists of setting all of the components of velocity to the velocity of the surface. This condition is most commonly used to model stationary walls, in which all of the components of velocity are set equal to zero.

A symmetry boundary condition is included to model boundaries where reflective symmetry is present. This boundary condition consists of no penetration and no shear conditions. The no penetration condition equates to a zero boundary condition for the velocity component normal to the surface. The no shear conditions equate to setting to zero the derivatives in the direction normal to the surface of the components of velocity transverse to the surface. This boundary condition acts as a mirror on the applied surface. Because of this fact, it is often used to reduce the overall size of the computational domain by cutting it in half.

The periodic boundary condition is implemented by replacing all equations on the boundary with the matching equations on the opposing boundary in the geometry. In problems of flowing fluids, application of this condition is modified slightly for pressure. In the case of pressure driven flow, the pressure is divided into two parts, the periodic portion of the pressure drop and the linear portion of the pressure drop.

$$P(x,y,z) = P_{periodic}(x,y,z) + \frac{\Delta P}{L}(L-x) \tag{7.1}$$

Here x is the direction normal to the periodic boundary , L is the distance between the periodic boundary and the opposing boundary, and ΔP is the pressure drop between the two boundaries. This description of the pressure is used in model geometries that are infinite in length and typically contain some repeating geometric feature such as an obstacle or a restriction.

The inflow and outflow boundary conditions are complimentary in that they are always used together in modelling flow geometries. The inflow boundary conditions consist of setting the velocity and stress profiles on the boundary. The profiles are found by solving the set of flow equations and stress equations on the boundary that are independent of the direction normal to the boundary. Like the periodic boundary

conditions, the flow equations for the inflow boundary conditions require an extra piece of information about the pressure drop in the direction normal to the boundary, the pressure drop per unit length. This quantity can be specified directly, or it can be used as a Lagrange multiplier to introduce an additional equation used to specify the flow rate Q through the inflow boundary,

$$Q = \int_{\Gamma} v_x dA \tag{7.2}$$

where Γ is the inflow boundary and x is the direction normal to the inflow boundary.

The outflow boundary conditions, or fully-developed boundary conditions, consist of specifying that the flow no longer varies in the direction normal to the boundary. These conditions equate to setting the derivatives of velocity with respect to the direction normal to the boundary to zero.

7.3 Discretization

7.3.1 Elements and Basis Functions

As discussed in Section 3.2, a mixed finite element method is used to discretize the set of equations describing the viscoelastic flow problem. Discretization of the volume of the physical geometry is performed using meshes of purely hexahedral elements or tetrahedral elements of the Lagrange type. No degenerate elements are employed. Continuous linear basis functions are used for representation of the pressure and velocity gradient interpolant variables. Continuous quadratic basis functions are used for the representation of the velocity variables. Discontinuous Galerkin basis functions are used to represent the stress variables.

The linear and quadratic isoparametric hexahedral elements are shown in Fig. 7-1. As is demonstrated in Section 3.2.2 for two-dimensional elements, the basis functions for the hexahedral element are constructed by taking products of the Lagrange polynomials, eq. 3.29, representing one-dimensional line elements lying along each of the three

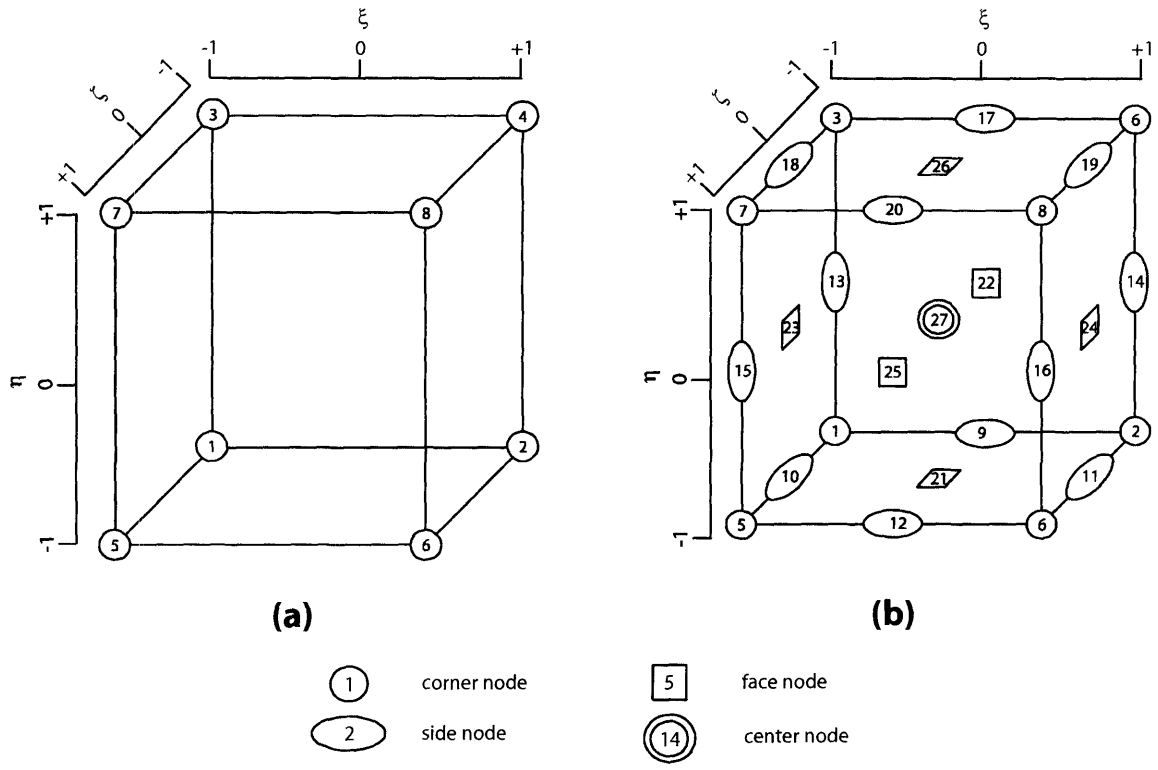


Figure 7-1: Hexahedral elements used to discretize the volume of a three-dimensional geometry. (a): 8-node linear hexahedral element; (b): 27-node quadratic hexahedral element. (ξ, η, ζ) isoparametric coordinate system is shown for each element.

isoparametric axes.

$$N_a(\xi,\eta,\zeta) = l_b^{n_{en}-1}(\xi)\, l_c^{n_{en}-1}(\eta)\, l_d^{n_{en}-1}(\zeta) \tag{7.3}$$

where n_{en} is the number of nodes along the one-dimensional edge (see Section 3.2.2 for further explanation) and b, c and d can be thought of as the nodes of three separate one-dimensional elements oriented along the ξ axis, the η axis, and the ζ axis. Continuous basis functions for the linear hexahedral element are then given by the expression

$$N_a(\xi,\eta,\zeta) = \frac{1}{8}(1+\xi_a\xi)(1+\eta_a\eta)(1+\zeta_a\zeta) \tag{7.4}$$

where (ξ_a, η_a, ζ_a) are the isoparametric coordinates of the a^{th} node shown in Fig. 7-1(a).

Basis functions for the quadratic hexahedral element are given by the expressions

$$N_a(\xi,\eta,\zeta) = \frac{1}{8}(\xi_a\xi)(1+\xi_a\xi)(\eta_a\eta)(1+\eta_a\eta)(\zeta_a\zeta)(1+\zeta_a\zeta) \tag{7.5a}$$

$$N_9(\xi,\eta,\zeta) = \frac{1}{4}\eta\zeta\left(1-\xi^2\right)(\eta-1)(\zeta-1) \tag{7.5b}$$

$$N_{10}(\xi,\eta,\zeta) = \frac{1}{4}\xi\eta(\xi-1)(\eta-1)\left(1-\zeta^2\right) \tag{7.5c}$$

$$N_{11}(\xi,\eta,\zeta) = \frac{1}{4}\eta\zeta(\xi+1)(\eta-1)\left(1-\zeta^2\right) \tag{7.5d}$$

$$N_{12}(\xi,\eta,\zeta) = \frac{1}{4}\eta\zeta\left(1-\xi^2\right)(\eta-1)(\zeta+1) \tag{7.5e}$$

$$N_{13}(\xi,\eta,\zeta) = \frac{1}{4}\xi\zeta(\xi-1)\left(1-\eta^2\right)(\zeta-1) \tag{7.5f}$$

$$N_{14}(\xi,\eta,\zeta) = \frac{1}{4}\xi\zeta(\xi+1)\left(1-\eta^2\right)(\zeta-1) \tag{7.5g}$$

$$N_{15}(\xi,\eta,\zeta) = \frac{1}{4}\xi\zeta(\xi-1)\left(1-\eta^2\right)(\zeta+1) \tag{7.5h}$$

$$N_{16}(\xi,\eta,\zeta) = \frac{1}{4}\xi\zeta(\xi+1)\left(1-\eta^2\right)(\zeta+1) \tag{7.5i}$$

$$N_{17}(\xi,\eta,\zeta) = \frac{1}{4}\eta\zeta\left(1-\xi^2\right)(\eta+1)(\zeta-1) \tag{7.5j}$$

$$N_{18}(\xi,\eta,\zeta) = \frac{1}{4}\xi\eta(\xi-1)(\eta+1)\left(1-\zeta^2\right) \tag{7.5k}$$

$$N_{19}(\xi,\eta,\zeta) = \frac{1}{4}\xi\eta(\xi+1)(\eta+1)\left(1-\zeta^2\right) \tag{7.5l}$$

$$N_{20}(\xi,\eta,\zeta) = \frac{1}{4}\eta\zeta\left(1-\xi^2\right)(\eta+1)(\zeta+1) \tag{7.5m}$$

$$N_{21}(\xi,\eta,\zeta) = \frac{1}{2}\eta\left(1-\xi^2\right)(\eta-1)\left(1-\zeta^2\right) \tag{7.5n}$$

$$N_{22}(\xi,\eta,\zeta) = \frac{1}{2}\zeta\left(1-\xi^2\right)\left(1-\eta^2\right)(\zeta-1) \tag{7.5o}$$

$$N_{23}(\xi,\eta,\zeta) = \frac{1}{2}\xi(\xi-1)\left(1-\eta^2\right)\left(1-\zeta^2\right) \tag{7.5p}$$

$$N_{24}(\xi,\eta,\zeta) = \frac{1}{2}\xi(\xi+1)\left(1-\eta^2\right)\left(1-\zeta^2\right) \tag{7.5q}$$

$$N_{25}(\xi,\eta,\zeta) = \frac{1}{2}\zeta\left(1-\xi^2\right)\left(1-\eta^2\right)(\zeta+1) \tag{7.5r}$$

$$N_{26}(\xi,\eta,\zeta) = \frac{1}{2}\eta\left(1-\xi^2\right)(\eta+1)\left(1-\zeta^2\right) \tag{7.5s}$$

$$N_{27}(\xi,\eta,\zeta) = \left(1-\xi^2\right)\left(1-\eta^2\right)\left(1-\zeta^2\right) \tag{7.5t}$$

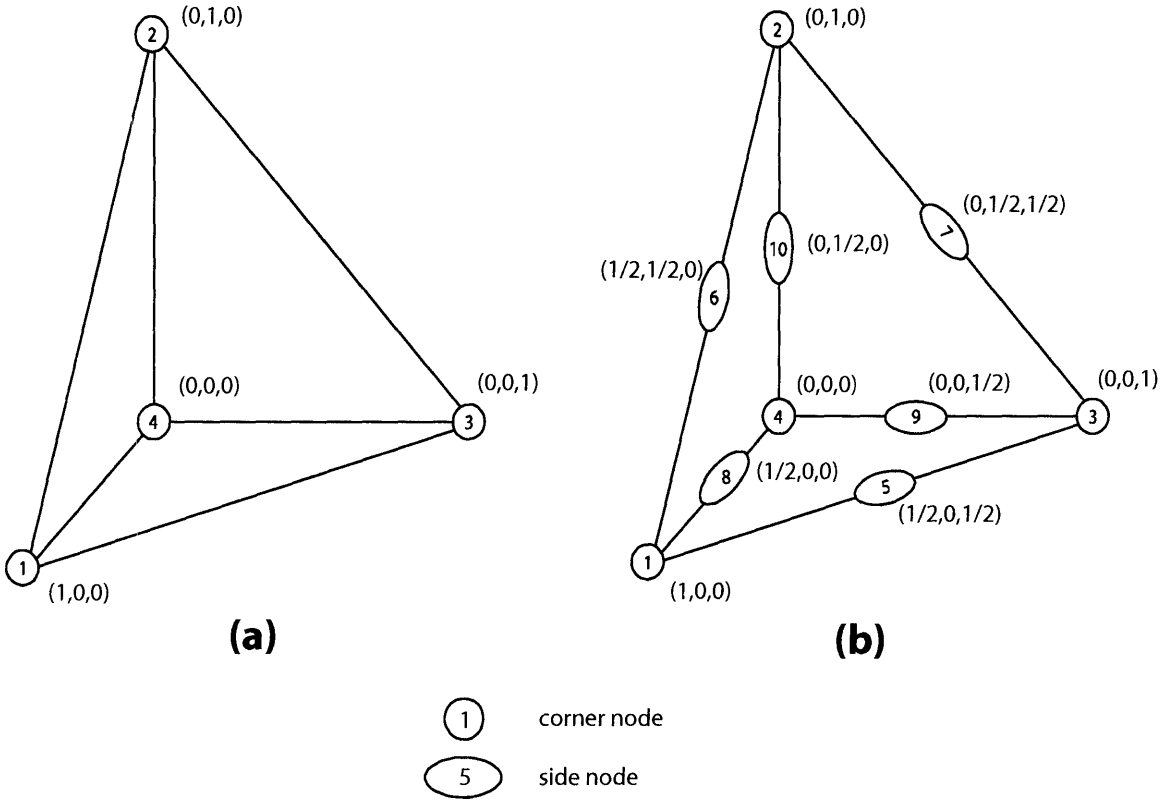


Figure 7-2: Tetrahedral elements used to discretize the volume of a three-dimensional geometry. (a): 4-node linear tetrahedral element; (b): 10-node quadratic tetrahedral element. Tetrahedral (r, s, t) isoparametric coordinates are shown for each node in the two elements.

where the node numbers correspond to those in Fig. 7-1 with a being the corner nodes 1 through 8.

The linear and quadratic isoparametric tetrahedral elements are shown in Fig. 7-2. Tetrahedral (r, s, t) isoparametric coordinates are shown for each node in the two

elements. Basis functions for the trilinear tetrahedral element are

$$N_1(r,s,t) = r \tag{7.6a}$$

$$N_2(r,s,t) = s \tag{7.6b}$$

$$N_3(r,s,t) = t \tag{7.6c}$$

$$N_4(r,s,t) = 1-r-s-t \tag{7.6d}$$

Basis functions for the triquadratic tetrahedral element are

$$N_1(r,s,t) = r(2r-1) \tag{7.7a}$$

$$N_2(r,s,t) = s(2s-1) \tag{7.7b}$$

$$N_3(r,s,t) = t(2t-1) \tag{7.7c}$$

$$N_4(r,s,t) = (1-r-s-t)(1-2r-2s-2t) \tag{7.7d}$$

$$N_5(r,s,t) = 4rs \tag{7.7e}$$

$$N_6(r,s,t) = 4st \tag{7.7f}$$

$$N_7(r,s,t) = 4t(1-r-s-t) \tag{7.7g}$$

$$N_8(r,s,t) = 4r(1-r-s-t) \tag{7.7h}$$

$$N_9(r,s,t) = 4rt \tag{7.7i}$$

$$N_{10}(r,s,t) = 4s(1-r-s-t) \tag{7.7j}$$

7.3.2 Mesh Generation

Mesh generation for the three-dimensional geometries is quite complex compared to its one- and two-dimensional counterparts. To accomplish this task ICEM, a mesh generator commercially available from ANSYS, is employed [1]. This mesh generation package was chosen from among the many commercial and freeware packages available because it is capable of generating meshes with the level of complexity beyond that of the modeling efforts presented here, allowing for growth without changing packages. In addition,

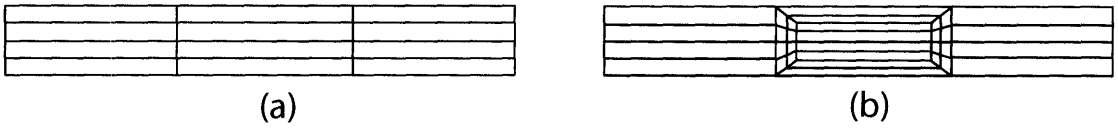


Figure 7-3: Example of o-grid refinement in a two-dimensional mesh used to localize the effects of mesh refinement. a) standard mesh with 4 elements in each section. b) mesh with o-grid refinment applied to central section, doubling the refinement.

considering the high level of complexity of geometries and meshes that can be created, the package is quite easy to use and manipulate. While both tetrahedral and hexahedral mesh generation modules are available within the ICEM package, meshes built for use in this work employ only the hexahedral meshing module, HEXA. The ICEM package has built in bandwidth minimization which is utilized here as well.

Due to the significant size of the three-dimensional finite element meshes, it is important that some form of local refinement be available within the meshing package. While ICEM is capable of generating meshes with local refinement using pseudo nodes, also known as hanging nodes, this has not been shown to be effective in the computation of viscoelastic fluid flows. ICEM also employs o-grid refinement to aid in localizing the effects of mesh refinement. Figure 7-3 is an example of the use of o-grid refinement in a two-dimensional mesh to double the mesh refinement in the vertical direction of the central section of the mesh. Note that the refinement applied to the central region of the mesh in has no impact on the mesh surrounding the central region. This technique can be used to concentrate elements around geometrical features such as corners and edges as well as provide means to transition mesh refinement between entry and exit regions such as those found in contraction and expansion geometries. Three-dimensional meshes generated within this work rely heavily on o-grid refinement to help reduce the overall size of the computational problem.

Further information on the ICEM mesh generation package is available on the web

at http://www.ansys.com.

7.4 Results of test problems

To test the validity of the three-dimensional finite element package, comparisons to solutions of trusted sources were performed. The geometries used for these tests are that of a cylindrical pipe and a square duct. Comparisons of the three-dimensional finite element package to the analytical solutions for the case of flow of a Newtonian fluid are first made. Comparisons of the new package to known solutions for the viscoelastic flow cases are also made, namely for a Giesekus model fluid.

7.4.1 Pipe Flow

The first test problem is that of pipe flow. This geometry was chosen since direct comparison to solutions from well tested two-dimensional package can be made. A schematic diagram of the three-dimensional test geometry is shown in Figure 7-4. Boundary conditions for the geometry are as follows. On the wall of the pipe, zero-velocity conditions are applied. On the left and right ends of the pipe, periodic boundary conditions are applied, namely all unknowns except for pressure are set equal to one another at the equivalent positions on the two boundaries and a uniform pressure gradient is specified in the direction of flow. The mesh used to discretize the geometry is shown in Fig. 7-5. It consists of 1344 hexahedral elements. An o-grid meshing structure is used to avoid degenerate elements on the centerline that would result from extruding a two-dimensional mesh in the azimuthal direction.

For the flow of a Newtonian fluid in a cylindrical pipe the analytical solution is well known [11]. The velocity profile as a function of flow rate is given by

$$v_x = \frac{2Q}{\pi R^2}\left[1 - \left(\frac{r}{R}\right)^2\right] \tag{7.8}$$

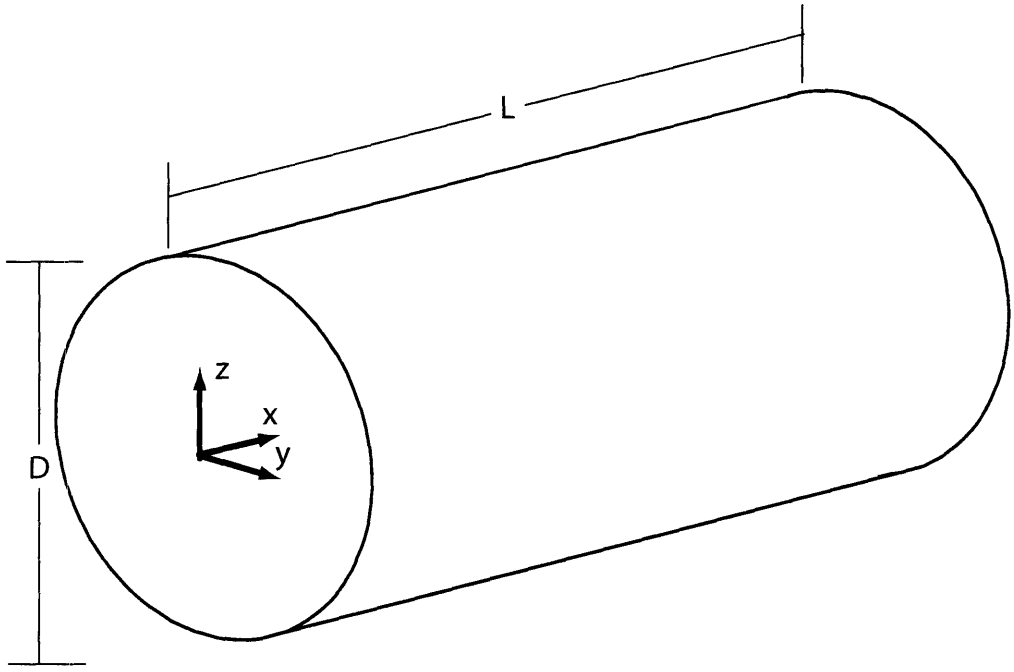


Figure 7-4: Schematic diagram of a pipe with circular cross-section. Flow is in the x direction. Inflow and outflow faces are in the yz plane. $L = 5$ is the length of the pipe used in the simulations. $D = 2$ is the diameter of the pipe used in the simulations. Note that Cartesian coordinates are used to describe the structure in three-dimensional space.

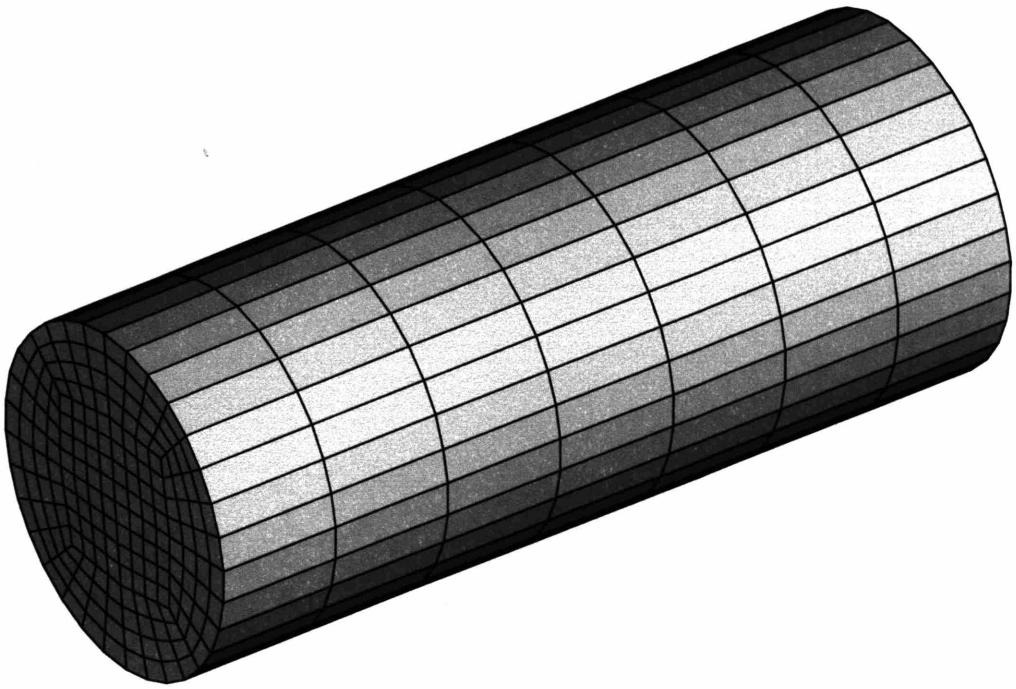


Figure 7-5: Mesh used for simulations of flow of a fluid in a pipe with cylindrical cross-section. The mesh contains 1344 hexahedral elements.

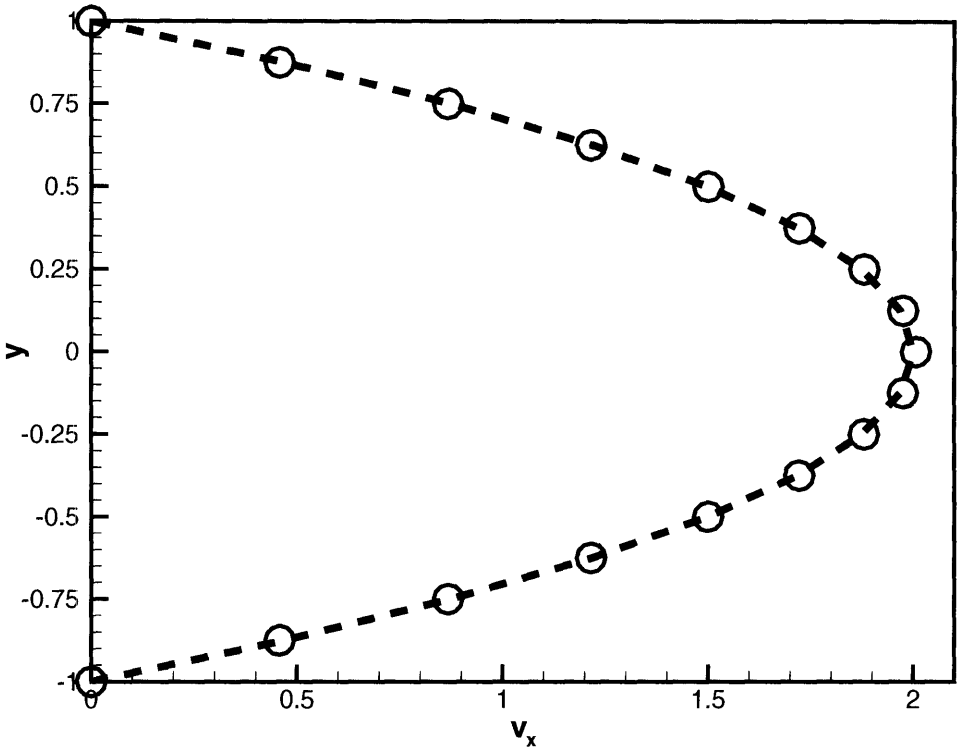


Figure 7-6: Comparison of the velocity field generated with the three-dimensional finite element package to the analytical solution for flow of a Newtonian fluid in a cylindrical pipe. - -: analytical solution; ◯: finite element solution.

where v_x is the velocity in the x direction, Q is the volumetric flow rate, R is the radius of the pipe, and $r = \sqrt{y^2 + z^2}$ is the radial location within the pipe.

Comparison of the flow field of the three-dimensional solution and the analytical solution is shown in Fig. 7-6. Excellent agreement between the three-dimensional package solution and the analytical method is found.

Since analytical solutions are not available for flows using more complicated viscoelastic constitutive equations, comparison of the three-dimensional package solution to a well tested and documented two-dimensional finite element package is used [16]. The mesh used in the two-dimensional finite element package is shown in Fig. 7-7. For these

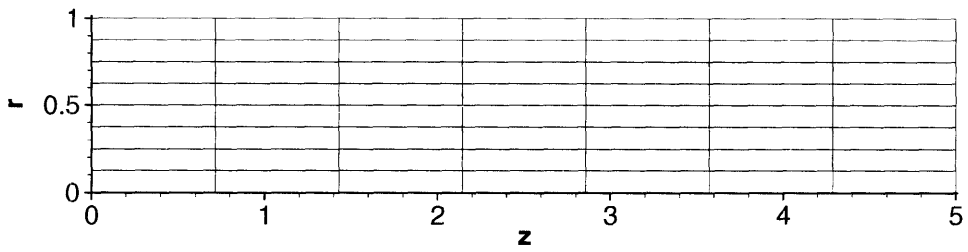


Figure 7-7: Mesh used for simulations of flow of a fluid in a pipe in the two-dimensional finite element method. The mesh contains 56 quadrilateral elements.

comparisons, the Giesekus model (eq. 2.26a) with $\beta = 0.5$, $\lambda = 1$, and $\alpha = 0.1$ is used to model the viscoelastic fluid. For De = 1.0, a comparison of the flow field found with the three-dimensional package and the two-dimensional package is shown in Fig. 7-8. A comparison of the shear and normal components of the stress tensor found with the two packages is shown in Fig. 7-9. As with the comparison of the Newtonian flow simulation to the analytical solution, excellent agreement is found between the axisymetric two-dimensional solver used in [16] and the new three-dimensional package.

7.4.2 Duct Flow

The second type of geometry used to test the three-dimensional solver is the flow through a duct. It is useful to think of duct flows as two distinct types, namely a duct with finite length between bounding walls in the transverse flow directions, Fig. 7-10 and a duct of infinite width in one of the two transverse dimensions, more commonly referred to as flow between two parallel plates, Fig. 7-11.

For the flow of a Newtonian fluid through a duct, the analytical solution is well known for the case of the where the width of the slit is infinite [27]. For this geometry, the

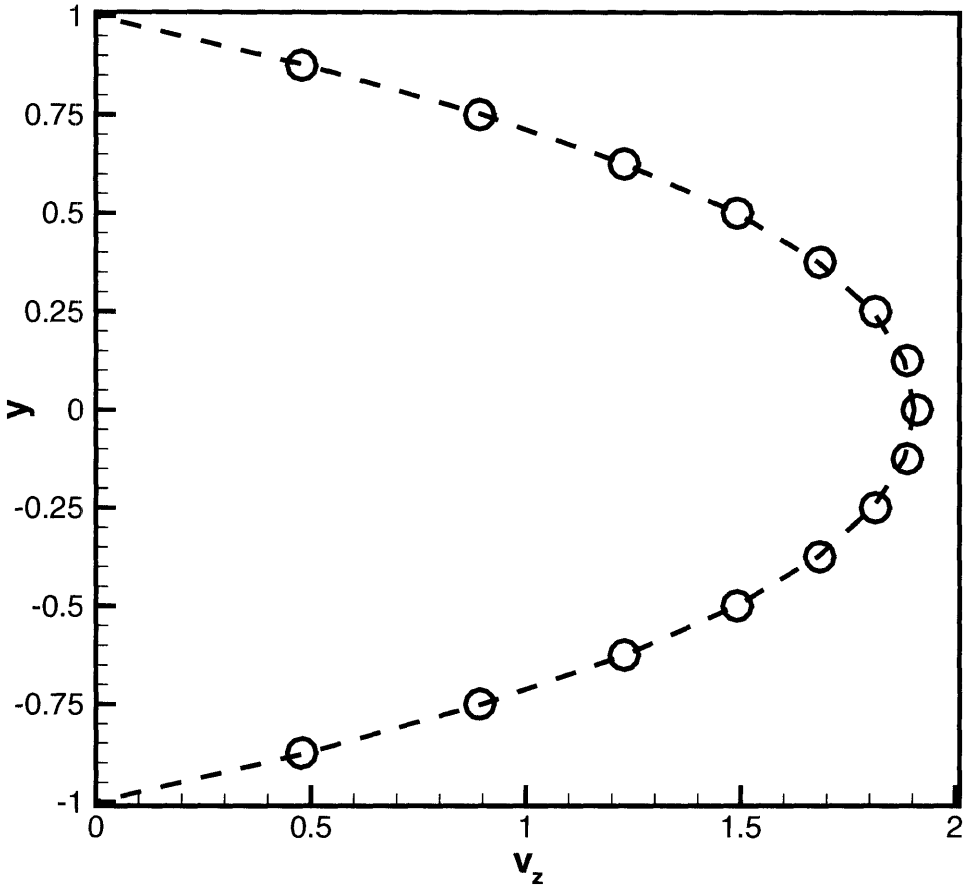


Figure 7-8: Comparison of the velocity field generated with the three-dimensional finite element package and the two-dimensional finite element package used in [16] for flow of a Giesekus model fluid in a cylindrical pipe. Parameters for the model fluid are $\beta = 0.5$, $De = 1.0$, and $\alpha = 0.1$. - -: 2-D axisymetric solution; ◯: 3-D solution.

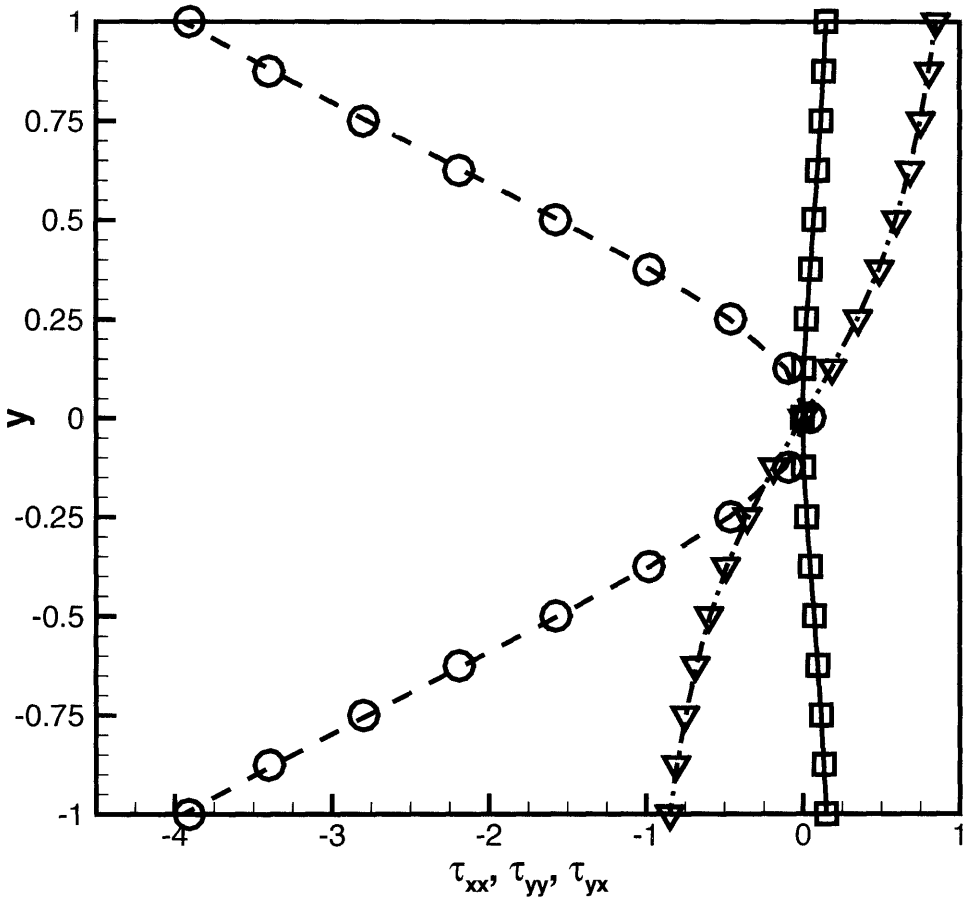


Figure 7-9: Comparison of the shear and normal components of the stress tensor generated with the three-dimensional finite element package and the two-dimensional finite element package used in [16] for flow of a Giesekus model fluid in a cylindrical pipe. Parameters for the model fluid are $\beta = 0.5$, De $= 1.0$, and $\alpha = 0.1$. Symbols represent the solution from the 3-D package, and lines represent the same from the 2-D package. - - and ◯: τ_{xx}; — and □: τ_{yy}; — · — and ∇: τ_{yx}.

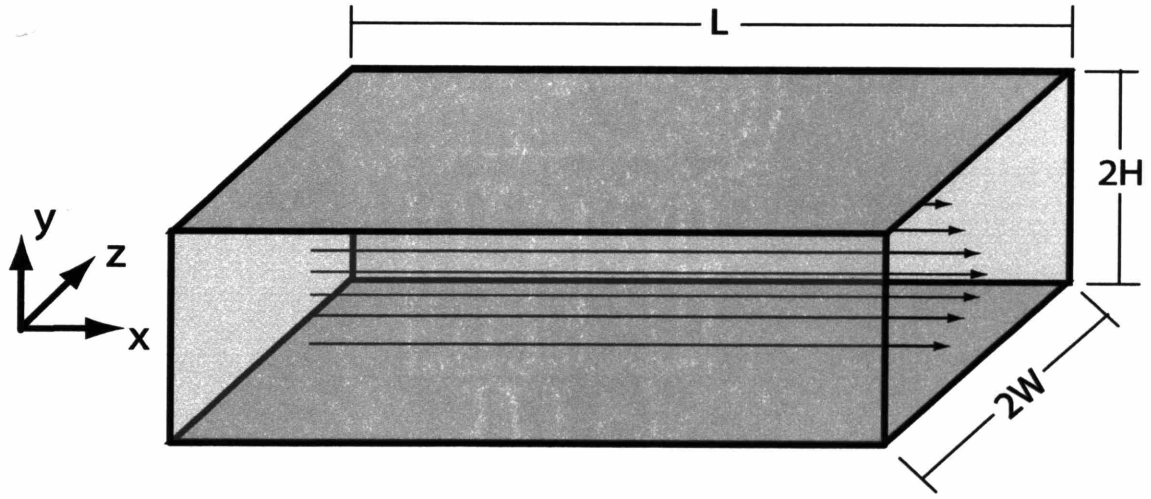


Figure 7-10: Schematic of flow in a rectangular duct. Flow is in the x direction. Inflow and outflow faces are in the yz plane. $L = 2$ is the length of the duct used in the simulations. $2H = 1$ is the height of the duct and $2W = 1$ is the width of the duct.

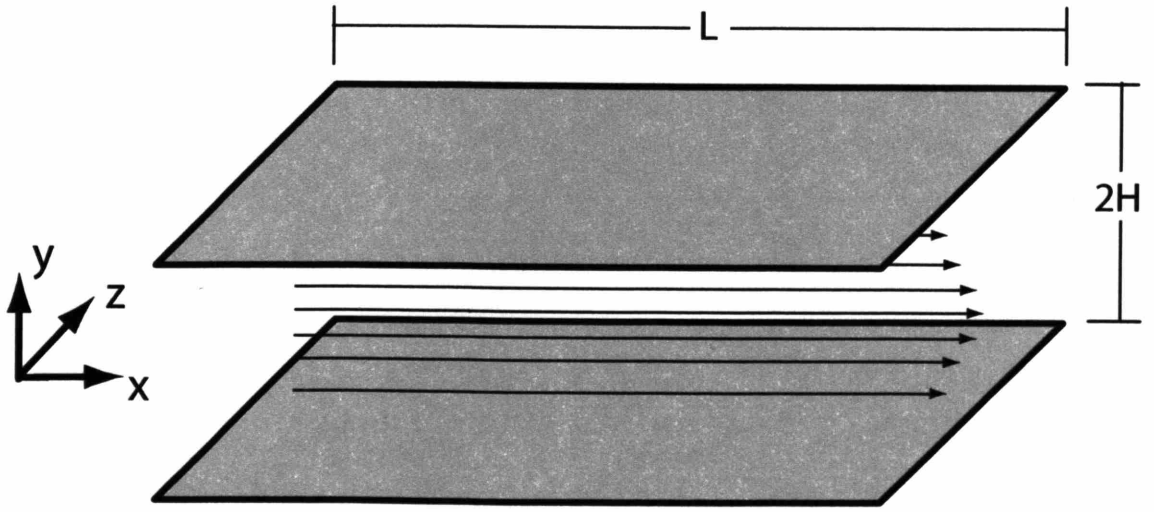


Figure 7-11: Schematic of flow between parallel plates. Flow is in the x direction. Inflow and outflow faces are in the yz plane. $L = 2$ is the length of the duct used in the simulations. $2H = 1$ is the height of the duct used in the simulations.

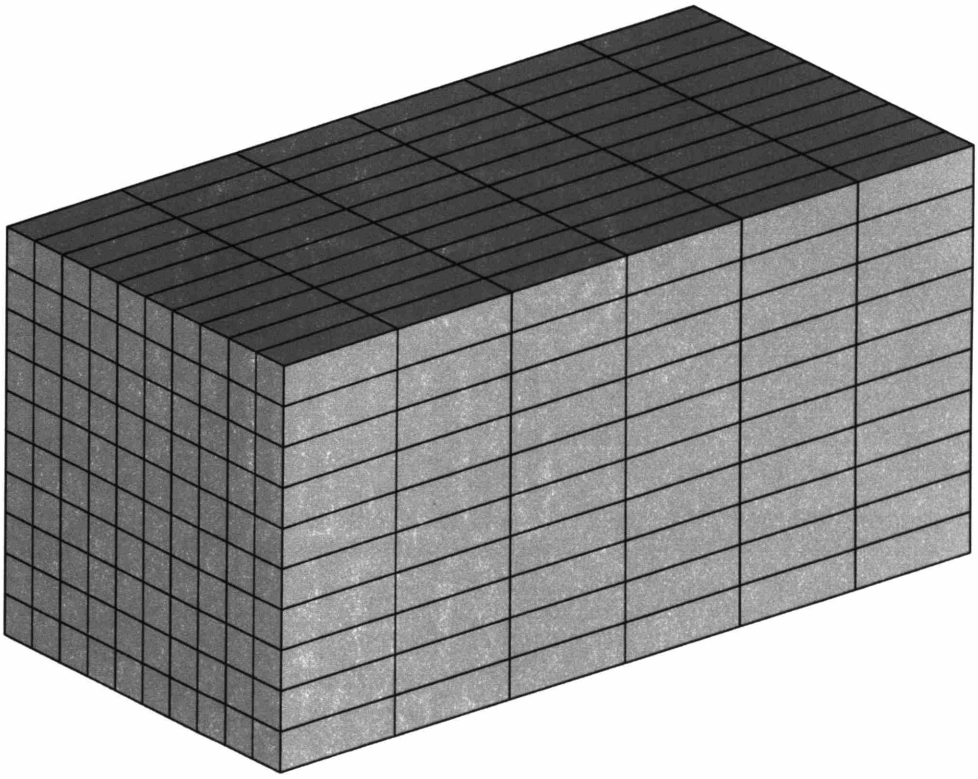


Figure 7-12: Mesh used for simulations of flow of a fluid in a duct with a square cross-section. The mesh contains 600 hexahedral elements.

velocity profile in the flow direction is given by

$$v_x = \frac{3q}{4H}\left[1 - \left(\frac{y}{H}\right)^2\right] \tag{7.9}$$

where q is the volumetric flow rate per unit width of the channel and H is the half height of the channel.

The mesh used for the duct flow calculations with the three-dimensional solver is shown in Fig. 7-12. The mesh has a square cross-section and contains a total of 600 hexahedral elements. To model the flow between two parallel plates, on the top and

bottom surfaces a zero velocity boundary condition is applied. On the left and right side surfaces periodic boundary conditions are applied with a pressure gradient of zero in the direction normal to the faces. On the front and back surfaces inflow and outflow boundary conditions are applied.

Comparison of the flow field of the three-dimensional solution and the analytical solution is shown in Fig. 7-13. As with the pipe flow, excellent agreement with the analytical solution is demonstrated.

An analytical series solution has been derived for flow of a Newtonian fluid through an infinitely long rectangular duct of cross-section $2a$ by $2b$ [85]. The velocity profile in the cross-section of the duct is given as

$$v_x(y,z) = \frac{16a^2}{\mu\pi^3}\left(-\frac{dp}{dx}\right)\sum_{i=1,3,5,\ldots}^{\infty}(-1)^{(i-1)/2}\left[1-\frac{\cosh(i\pi z/2a)}{\cosh(i\pi b/2a)}\right]\frac{\cos(i\pi y/2a)}{i^3} \tag{7.10}$$

For the square duct case modeled here, $2a = 2b = 1$. Boundary conditions for simulation of this flow geometry consist of zero velocity conditions on the top, bottom, left, and right surfaces. Inflow and outflow boundary conditions are used on the front and back surfaces, respectively.

Comparison of the flow field of the three-dimensional solution and the analytical solution is shown in Fig. 7-14. Again, excellent agreement with the analytical solution is found.

For comparison of flow in the square duct of viscoelastic fluids modeled with constitutive equations yielding non-vanishing second normal stress differences, as with the Giesekus model, analytical solutions are not available. It is well documented in the literature that flows of these fluids in the square duct exhibit a secondary flow that is mirrored on four planes of symmetry [86], [26]. The secondary flow computed by Xue et al. [86] for a single quadrant in the square duct flow is shown in Fig. 7-15. The streamtraces of the secondary flow found in the three-dimensional simulation are shown in Fig. 7-16. The presence of the symmetry of the secondary flow along the $y = 0.5$,

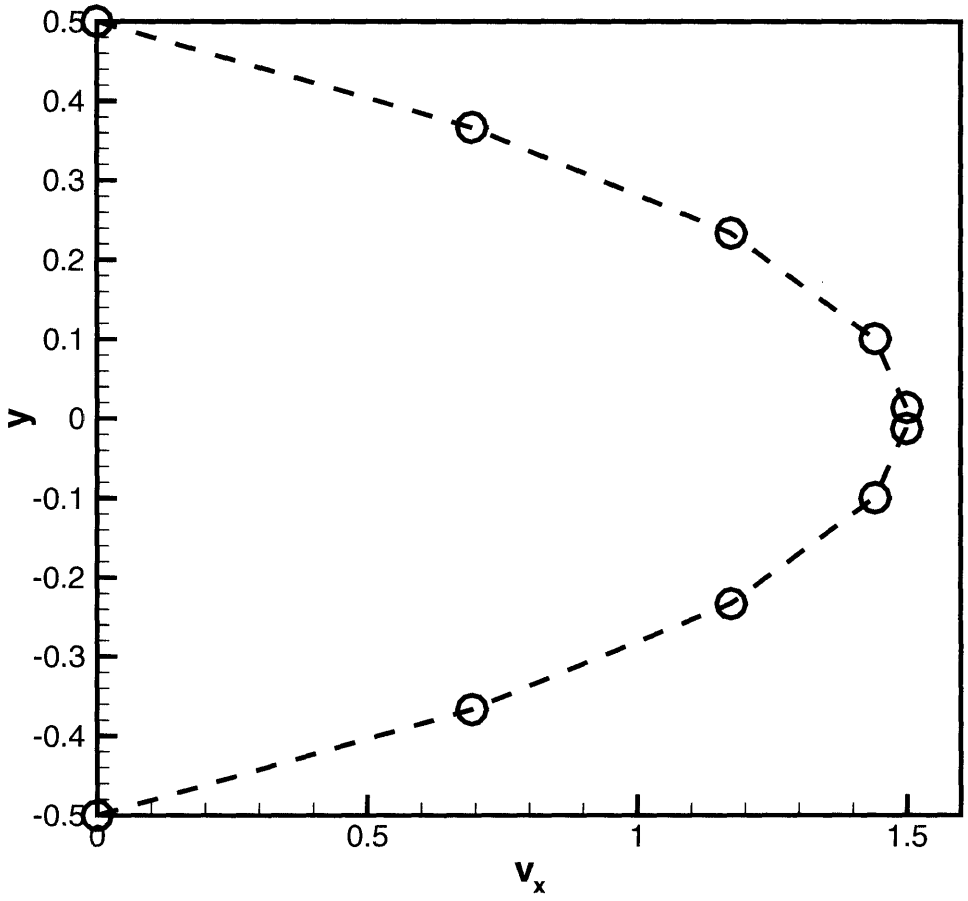


Figure 7-13: Comparison of the velocity field generated with the three-dimensional finite element package to the analytical solution for flow of a Newtonian fluid between parallel plates. - -: analytical solution; ◯: finite element solution.

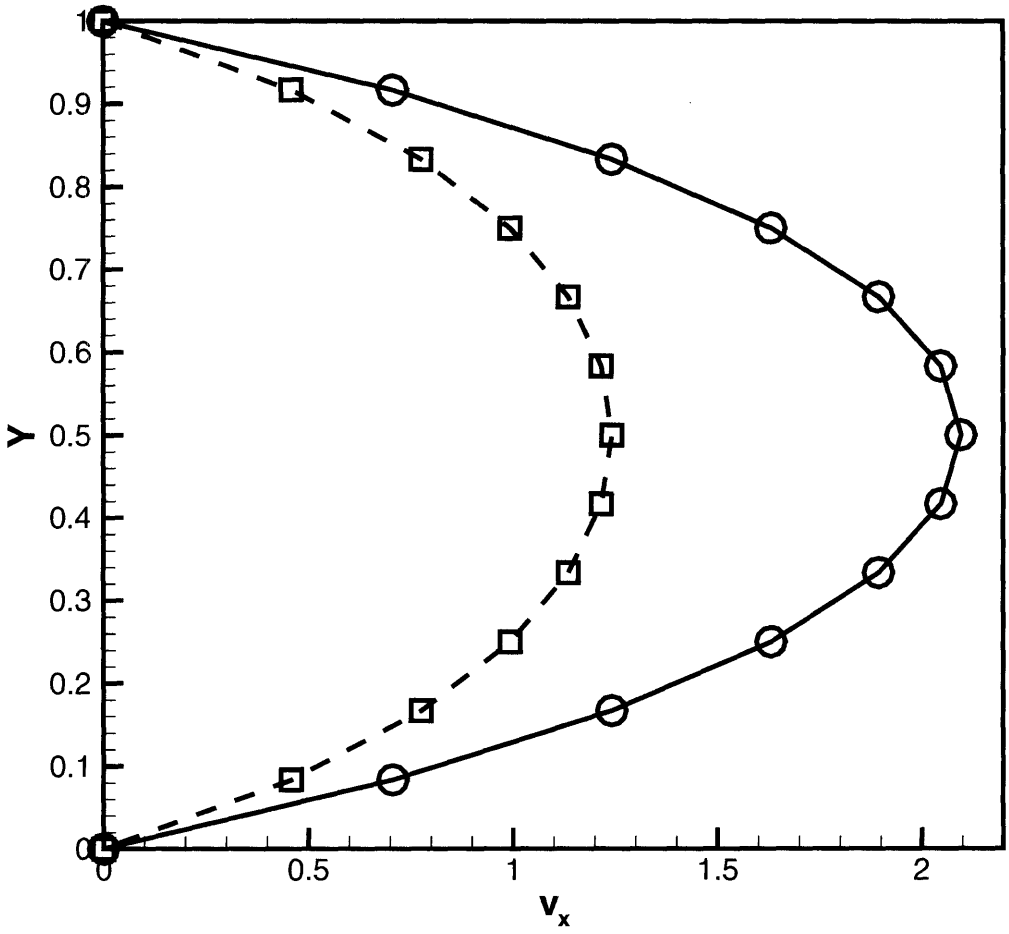


Figure 7-14: Comparison of the velocity field generated with the three-dimensional finite element package to the analytical solution for flow of a Newtonian fluid in a duct with a square cross section. - -: analytical solution at $z = 1/3$; □: finite element solution at $z = 1/3$;—: analytical solution at $z = 1/2$; ◯: finite element solution at $z = 1/2$.

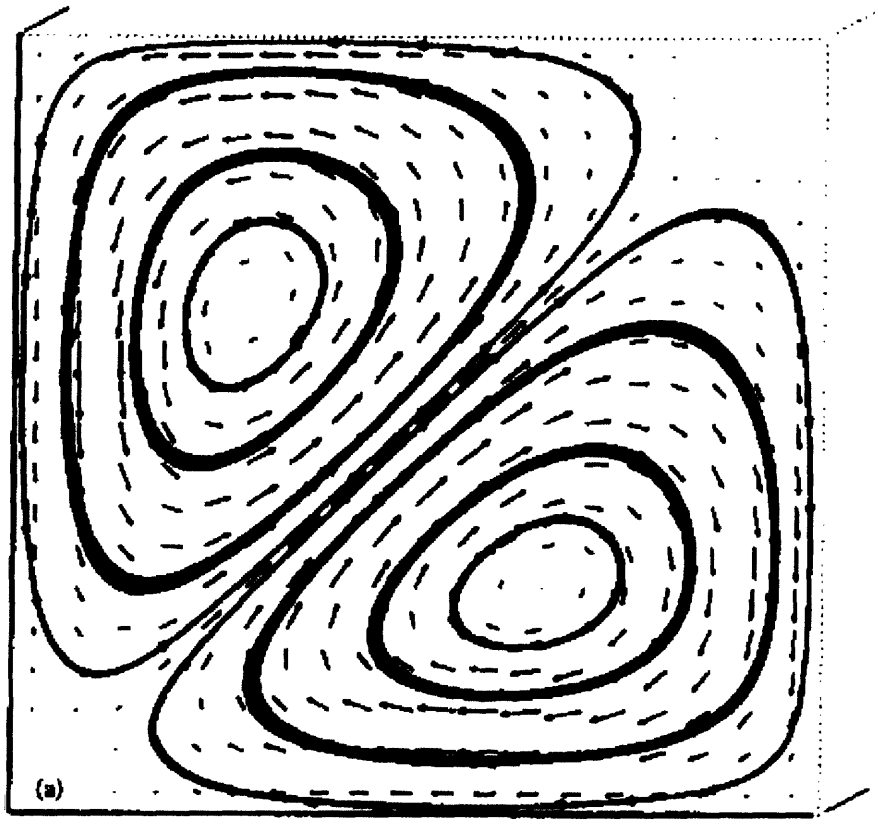


Figure 7-15: Secondary flow generated in the flow of an MPTT fluid in a square duct. The image shows streamtraces in a single quadrant of the cross-section of the duct. Model parameters as reported in [86] are $\rho = 1$, $\eta_{m0} = 1$, $\lambda_{\max} = 0.01$, $\epsilon = 0.1$, $\xi = 0.2$, $n = 0.65$, and $\beta = 1$. Figure reproduced from [86].

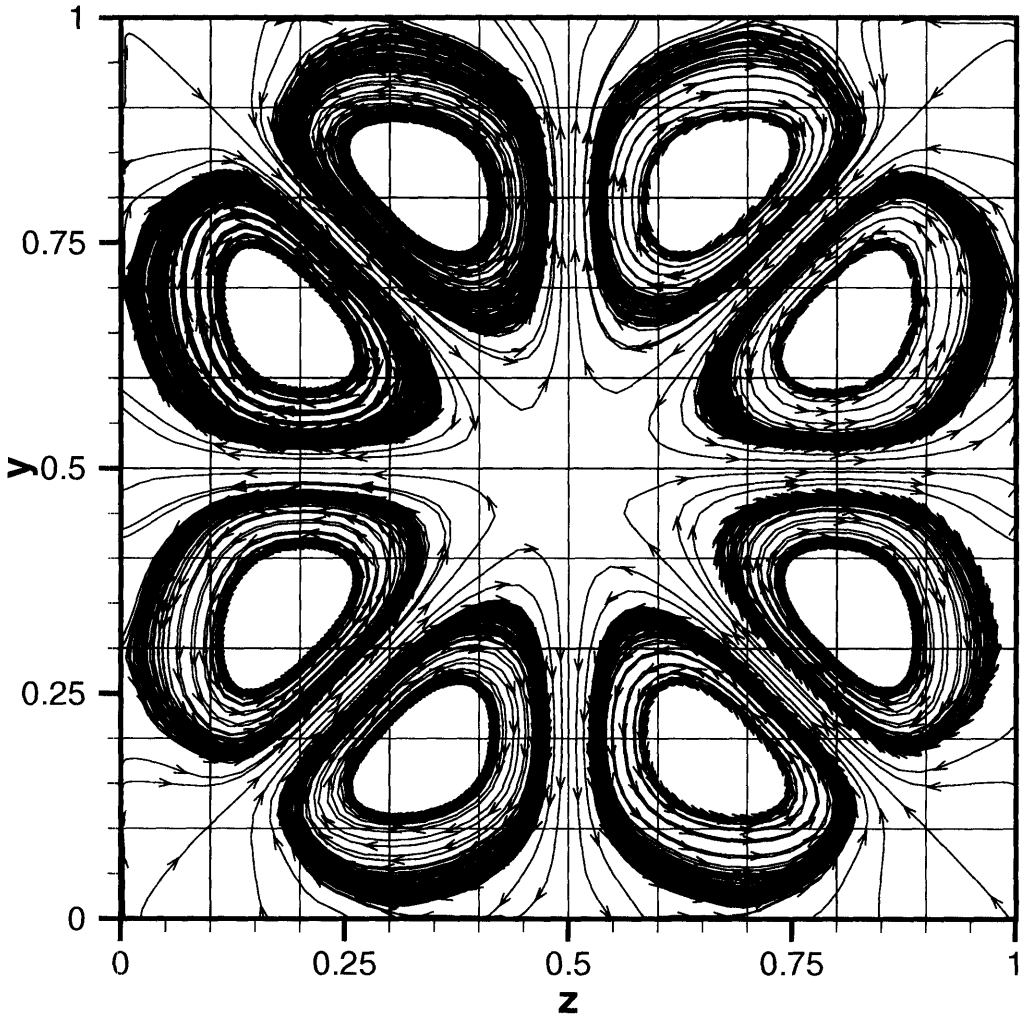


Figure 7-16: Secondary flow generated from flow of a Giesekus fluid with $\beta = 0.5$, De $= 1.0$, and $\alpha = 0.1$ in a square duct of $2H = 1$, $2W = 1$, and $L = 2$. Symmetry clearly present along $y = 0.5$, $z = 0.5$, $y = z$, and $y = 1 - z$ planes.

$z = 0.5$, $y = z$, and $y = 1 - z$ planes is clearly present.

7.5 Parallel method

The parallel method used in the three-dimensional solver is a direct extension of the two-dimensional parallel viscoelastic flow solver used in [16]. Details of the method are given in Section 3.5 in a form independent of the dimension of the problem. In addition, the subproblem describing the velocity gradient interpolant is solved by using the Conjugate Gradient method. Each component of $\mathbf{G}$ is independent of the other components, and therefore the components are solved for separately, further reducing the overall size of the calculation. This results in a set of calculations that requires an insignificant amount of time when compared to the solution of the Stokes problem. The relative time to reach solution in each subproblem is provided below.

Testing of the performance of the three-dimensional solver was performed on a parallel cluster of 32 Dell 2650 rack-mounted servers. The machines each contain dual Pentium Xeon processors with 512 KB L2 cache and 2 GB RAM. Each processor has a clock speed of 2.8 GHz. The Linux operating system with kernel 2.4.1 was used on each machine. The parallel environment consisted of LAM version 6.5.9 and PETSC version 2.1.6. The machines were interconnected with an Extreme Networks Black Diamond 6808 switch containing 40 fully non-blocking 1 Gb copper ethernet ports and both primary and secondary management modules to provide adequate switching fabric to ensure no greater than 1:1 subscription ratio for each port.

While the ultimate goal of the development of the parallel solver was to reduce the total time needed to complete the calculation as compared with the serial analog, it is of interest to measure how efficiently the resources of the parallel machine are being used. Armed with efficiency measurements, one is much more apt to predict correctly how the solver will perform for a wide range of problem sizes. A number of metrics exist for the measurement of the performance and efficiency of parallel solvers. The most relavant to

the current investigation are as follows.

The speedup metric is often used to measure the efficiency of the size of the parallel machine. Speedup $S = t_1/t_p$ is the ratio of the time to complete the calculation on one processor, t_1, divided by the time to complete the calculation on P processors, t_p given a fixed problem size. Efficiency of the parallel method is given by $E = S/P$. An ideal parallel method would have a linear speedup with a slope of one where doubling the number of processors would result in time decreasing by a factor of one half and an efficiency of one for any number of processors. Since the problem size is held constant, as the number of processors increases, the problem on each processor will decrease. In terms of the physical geometry, as the domain is broken up into smaller and smaller pieces, the relative size of the bordering area between the pieces increases. This in turn necessitates greater and greater amounts of communication between processors. Eventually the time needed for communication between processors will become much more significant than the amount of time spent on each processor's calculations, consistent with Amdahl's law [5], [68]. This occurence is manifested in the speedup metric when doubling the amount of processors results in the time decreasing by less than a factor of one half and in the efficiency metric when significant decrease from unity is seen with increasing number of processors. Because of this fact, the speedup and efficiency metrics are highly dependent on the hardware used to test the parallel method.

Measurements of speedup and efficiency are given in Fig. 7-17 for a highly refined square duct mesh of 5,000 elements with 450,000 unknowns. For a problem of this magnitude, the parallel three-dimensional solver exhibits linear speedup and efficiency of unity with up to 8 processors in the parallel machine. For 16 processors the speedup and efficiency are no longer ideal, and for 32 processors little gain in speed is realized as compared to the 16 processor case. The optimal machine size for this problem is therefore 8 processors, though some overall speed increase is seen for up to 32 processors.

A more refined set of metrics used by Gustafson [38] for determining the performance and efficiency of the parallel method are defined as above but with a problem size per

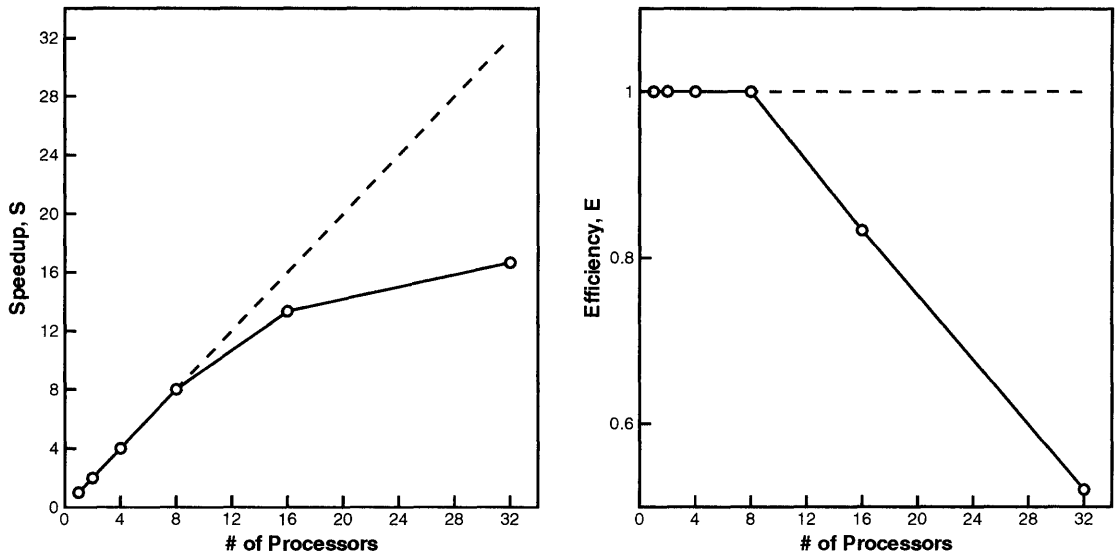


Figure 7-17: Speedup and efficiency of three-dimensional parallel method for flow of a Giesekus fluid ($De = 1.0$, $\beta = 0.5$, $\alpha = 0.1$) in a square duct. The mesh consists of 5,000 elements and a total of 450,000 unknowns.

processor that is held constant. Fig. 7-18 shows the Gustafson efficiency for numbers of unknowns per processor of 14115, 44788, and 177088. Since it is quite difficult to fix exactly the number of unknowns per processor, the efficiency, E, for the P processor parallel machine is multiplied by the ratio of number of unknowns per processor for $P = n$ to the number of unknowns per processor for $P = 1$. For each of the two smaller problems, the Gustafson efficiency appears to asymptotically approach a single value for large n. This large-n efficiency appears to decrease slightly with increasing problem size. Due to limitations of the hardware, the largest probem could not be run on a machine greater than 8 processors.

Since the solver used here for the solution of the viscoelastic flow is comprised of multiple decoupled subproblems, it is instructive to consider the relative amount of time spent on each of the subproblem. For the square duct mesh of 5000 elements, the relative amount of time needed to compute the solution of each subproblem is shown in Table

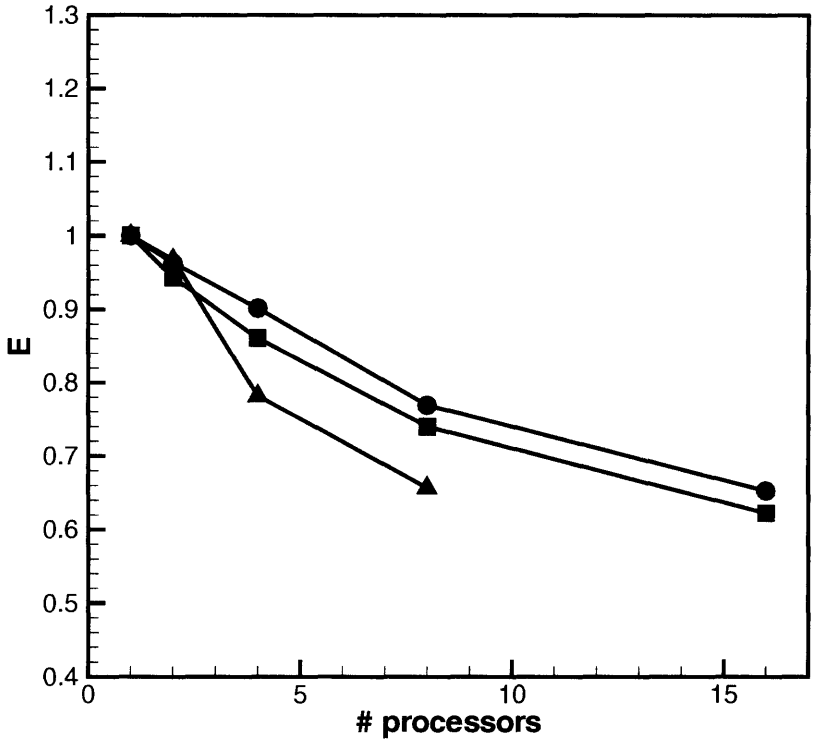


Figure 7-18: Gustafson efficiency as a function of the number of processors in the parallel machine for the three-dimensional finite element solver. Unknowns per processor: ○– 14115, □– 44788, Δ– 177088.

7.1. Clearly the amount of time needed to solve the Stokes problem greatly outweighs that of the velocity gradient interpolant and the stress equation. While this is certainly the case for any continuum stress equation, polymer stresses described by more advanced methods such as Brownian Dynamics and kinetic theory will undoubtedly require more significant amounts of computational time relative to the Stokes problem [76].

	v,p	G	τ
average iterations	40	20x9	-
time/iteration (seconds)	0.5	0.001	-
% of total time	95%	<1%	5%

Table 7.1: Relative time needed to reach solution for the momentum/mass continuity, velocity gradient interpolant, and stress subproblems in the parallel three-dimensional finite element package. Measurements are made for the calculation of the flow of a Giesekus fluid in a square duct with 5000 elements.

7.6 Summary

A three-dimensional finite element package for the computation of confined viscoelastic flows was detailed in this chapter. The package employs the DEVSS-G formulation and uses a time-dependent method to decouple the stress, flow, and velocity gradient interpolant equations. While the Giesekus equation is the constitutive equation implemented in the package, the decoupled form of the equations allows for easy implementation of any differential constitutive equation. The package is designed to handle a wide range of geometries by including a number of different boundary conditions.

Detailed information concerning the elements and basis functions used in the package are included for completeness of description of the method. Meshing to discretize the physical domain is accomplished using the ICEM mesh generator commercially available from ANSYS. ICEM employs o-grid refinement to localize the effects of mesh refinement, reducing the overall size of the mesh.

The accuracy of the method has been demonstrated using a number of pipe and duct flows with comparison to analytical and computational results. Comparison of the solutions with established results for flow of a Newtonian fluid and a Giesekus fluid show excellent agreement.

Parallelization of the method is a direct extension of the two-dimensional method in [16]. The measurements of the speedup and efficiency of the method demonstrate that application of this parallelization technique is effective for the three-dimensional viscoelastic flow calculation.

Chapter 8

Flow Across a Periodic Array of Cylinders

To test the three-dimensional finite element package on a problem of complex flow, the periodic, linear array of cylinders was chosen. Flow in this geometry exhibits a combination of shear, predominately in the gaps between the cylinder and the top and bottom walls, and elongation, predominately in the region immediately upstream and downstream of the cylinder. The periodic nature of the geometry allows for more mesh to be concentrated in regions of large gradients rather than long upstream and downstream sections of the geometry as is necessary in entry and exit flow simulations such as the axisymmetric 4:1:4 contraction-expansion flow geometry that is modeled in chapter 6.

Much work has been accomplished in characterizing the flows over the full range of separation distances between the cylinders in the periodic array ([6], [54], [55], [57], [73]). The most significant body of work on this problem is for the inter-cylinder spacing of 2.5, as performed by Liu [55] and Smith et al. [73]. Liu performed extensive experimental observations for the flow of a polyisobutylene Boger fluid through a periodic, linear array of cylinders for cylinder spacings of 2.5, 6 and ∞, the isolated cylinder case. Transitions from a two-dimensional flow to a three-dimensional, time-dependent flow were identified for critical values of the Weisenberg number that depended on the cylinder

spacing. Smith et al. performed linear stability analysis by applying three-dimensional perturbations to a two-dimensional base-state flow. Using the Oldroyd-B model, they were able to capture the onset of the instability for the periodic array of cylinders over a range of cylinder spacings. Their calculations of the critical Weisenberg number for the onset of the instability as well as the wave number of the instability agree quite well with Liu's findings for the 2.5 case.

The goal of this chapter is to study the flow of a Newtonian fluid and the Oldroyd-B model in the periodic, linear array of cylinders with cylinder spacing of 2.5 and characterize the effects of variations in the width of the geometry using a full three-dimensional finite element simulation. Section 8.1 presents the geometry, governing equations, and numerical methods used in the simulations. Section 8.2 presents the results of the simulations beginning with the Newtonian results in Subsection 8.2.1 and then the Oldroyd-B model results in Subsection 8.2.2. Conclusions are given in Section 8.3.

8.1 Problem Description

8.1.1 Geometry

The geometry considered here is a periodic, linear array of cylinders designed to match the experimental system used in [55], a schematic diagram of which is shown in Fig. 8-1. All physical dimensions are non-dimensionalized with the cylinder radius, R_c. The axis of the cylinder is centered between the top and bottom walls of the channel. While geometries with various distances between cylinders were considered by Liu, only geometries with a separation length $L_c = 2.5R_c$ between the cylinder centers are considered here. The height of the channel is $2H = 4R_c$. The width of the channel is $W = 4R_c$ for the majority of the calculations, though geometries of $W = 2R_c$ and $W = 3R_c$ are also simulated. The periodic computational domain is from the center of gap fore and aft of one cylinder with a length of $L_d = L_c$, denoted by the dashed lines in Fig. 8-1. The origin of the Cartesian coordinate system used to describe the geometry is located

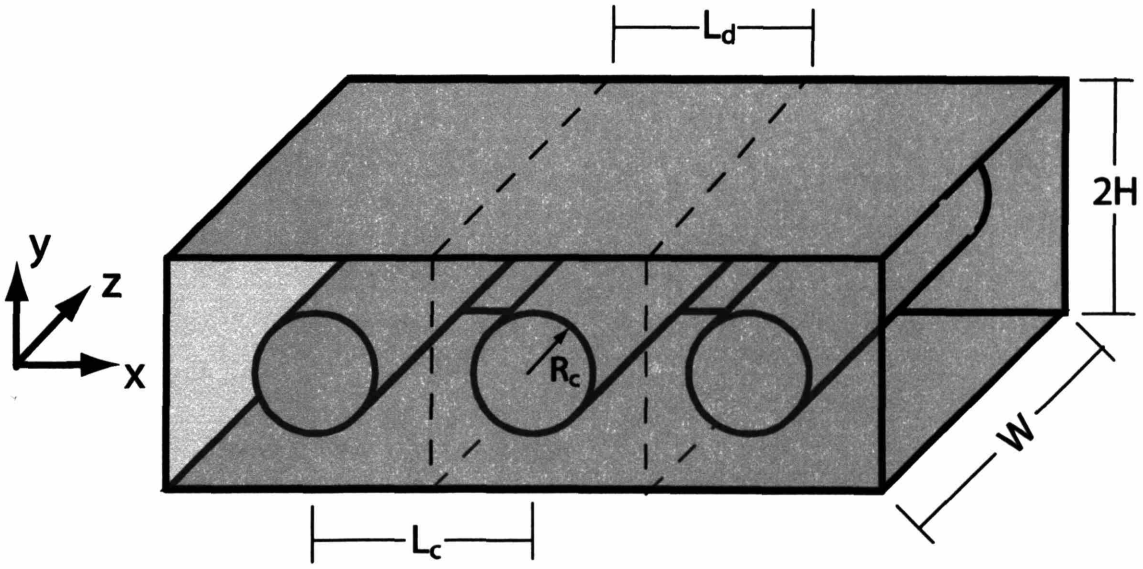


Figure 8-1: Schematic diagram of the periodic array of cylinders. The distance between the cylinder centers is $L_c = 2.5R_c$. The height of the channel is $2H = 4R_c$. The axis of the cylinder is centered between the top and bottom walls of the channel. The width of the channel is $W = 4R_c$ for the majority of the calculations, though geometries of $W = 2R_c$ and $W = 3R_c$ are also simulated. The periodic computational domain is from the center of gap fore and aft of one cylinder with a length of $L_d = L_c$, denoted by the dashed lines. The flow in the channel is in the positive x direction.

on the axis of the cylinder halfway between the ends of the cylinder. The flow in the channel is in the positive x direction.

8.1.2 Governing equations

Flow of the fluid through the periodic, linear array of cylinders is governed by the equations of momentum conservation, mass continuity, and equations describing the relationship between polymer stress and deformation of the fluid. To model the heterogeneous complex flow, the DEVSS-G formulation for the momentum and mass continuity equations is utilized. In the DEVSS-G formulation, the momentum equation employs a stress splitting to take the form

$$\nabla^2 \mathbf{v} - \nabla p - \nabla \cdot \left\{ (1-\beta) \left\{ \mathbf{G} + \mathbf{G}^T \right\} \right\} - \nabla \cdot \boldsymbol{\tau}_p = \mathbf{0} \tag{8.1}$$

where $\mathbf{v}$ is the fluid velocity, β is the solvent to zero-shear-rate viscosity ratio, p is the pressure, $\mathbf{G}$ is the velocity gradient tensor, $\boldsymbol{\tau}_p$ is the extra stress, and ∇ is the gradient operator. The mass continuity equation is given by,

$$\nabla \cdot \mathbf{v} = \mathbf{0} \tag{8.2}$$

A least squares minimization is used to compute the velocity gradient,

$$(\nabla \mathbf{v} - \mathbf{G})^2 = \mathbf{0} \tag{8.3}$$

The Oldroyd-B fluid model is chosen the model the polymer extra stress of the MIT Boger fluid used in [54]. This model was used successfully to describe the dynamics of this polymer system by Smith et al. [73] in the periodic, linear array of cylinders. The Oldroyd-B fluid model is given by,

$$\boldsymbol{\tau}_p + \mathrm{De} \boldsymbol{\tau}_{p(1)} = -(1-\beta) \left[\mathbf{G} + \mathbf{G}^T \right] \tag{8.4}$$

where De $= \lambda_0 \dot{\gamma}$ is the Deborah number, $\lambda_0 = \Psi_{10}/2\,(\eta_0 - \eta_s)$ is the zero-shear-rate polymer relaxation time, and $\beta = \eta_s/\eta_0$ is the solvent to total viscosity ratio. To match the zero-shear-rate rheological properties of the MIT Boger fluid used in [55] and modeled in [73] the parameters are chosen as $\lambda_0 = 0.3\ s$ and $\beta = 0.67$.

Boundary conditions applied to the boundaries of the computational domain shown in Fig. 8-1 are specified as follows. To model an array of cylinders that repeats indefinitely in the x direction, periodic boundary conditions are applied on the inflow and outflow boundaries. No slip boundary conditions are specified on the solid walls oriented in the xz plane and on the cylinder surface. On the walls bounding the computational geometry in the yz plane, two different conditions are applied. To model a geometry bounded in the z direction, no slip boundary conditions are applied on these walls. To model a geometry that is infinite in the z direction, periodic boundary conditions are applied on these walls.

8.1.3 Numerical method

The Galerkin finite element method is used to discretize the unknowns for the DEVSS-G equations, namely $\mathbf{v}$, p, and $\mathbf{G}$, and the discontinuous Galerkin method is used for unknowns for the constitutive equations describing the polymer stress, namely $\boldsymbol{\tau}_p$. The mesh used to discretize the three-dimensional physical domain is shown in figure 8-2. This structured mesh consists of 14420 hexahedral elements. The meshes used to model the $W = 2R_c$ and $W = 3R_c$ geometries are identical to the $W = 4R_c$ mesh in all but the z coordinate. Meshes identical to each other were chosen to better compare the flow fields arising the in different geometries. Increased resolution in the z direction with decreasing width W is necessary if flow structures periodic in the z direction are to be resolved.

For comparison purposes, simulation of the periodic, linear array of cylinders assuming symmetry about the mid-plane $y = 0$ were also performed. The mesh used for these simulations, Fig. 8-3, is identical to the upper half of the mesh shown in Fig. 8-2. The

mesh consists of 7210 hexahedral elements.

The mesh used in the two-dimensional simulations, in which variations in the z direction are neglected and v_z is assumed to be zero, is shown in Fig. 8-4. This mesh is simply a cross-section of the three-dimensional mesh. It consists of 412 quadrilateral elements.

The index one set of DAE's resulting from the finite element discretization of the governing equations is then converted into a set of first order ODE's by using the operator splitting technique described in [77]. Once written in this form, the set of equations can be decoupled and solved in separate sub-steps by employing an explicit time integration technique, in which the equations describing the flow, eqs. 8.1-8.3, become constraints on the evolution equations describing the polymer extra stress, eq. 8.4. The flow equations are further decoupled by first solving equations 8.1 and 8.2 and then using the new velocity fields to compute the $\mathbf{G}$ interpolants from equation 8.3. The momentum and continuity equations are preconditioned and then solved with a Krylov iterative technique as in [16] and further described for the three-dimensional method in Section 7.5. Unlike the work of Caola et al. [16] in which BiCGStab was used as the iterative solver, here GMRES is employed for its improved stability and robustness. Also unlike the work of Caola et al., a separate solution technique can now be employed to solve for the least squares interpolation for $\mathbf{G}$ since equation 8.3 is now decoupled from the flow equations. Here the Conjugate Gradient method is employed for this symmetric, positive definite set of equations. All of the sub-problems are solved by using parallel versions of the Krylov solvers and the domain decomposition CHACO to optimally distribute the equations among the nodes in the parallel machine. Further information on this method can be found in [16]. The computations were carried out on a parallel cluster of 32 Dell 2650 PowerEdge servers each with dual 2.8 GHz Intel Pentium Xeon processors and 2 gigabytes of memory. The intercommunication between machines was carried out via an Extreme Black Diamond 6808 switch with gigabit copper NIC's and adequate bandwidth for a fully non-blocking and non-oversubscribed architecture.

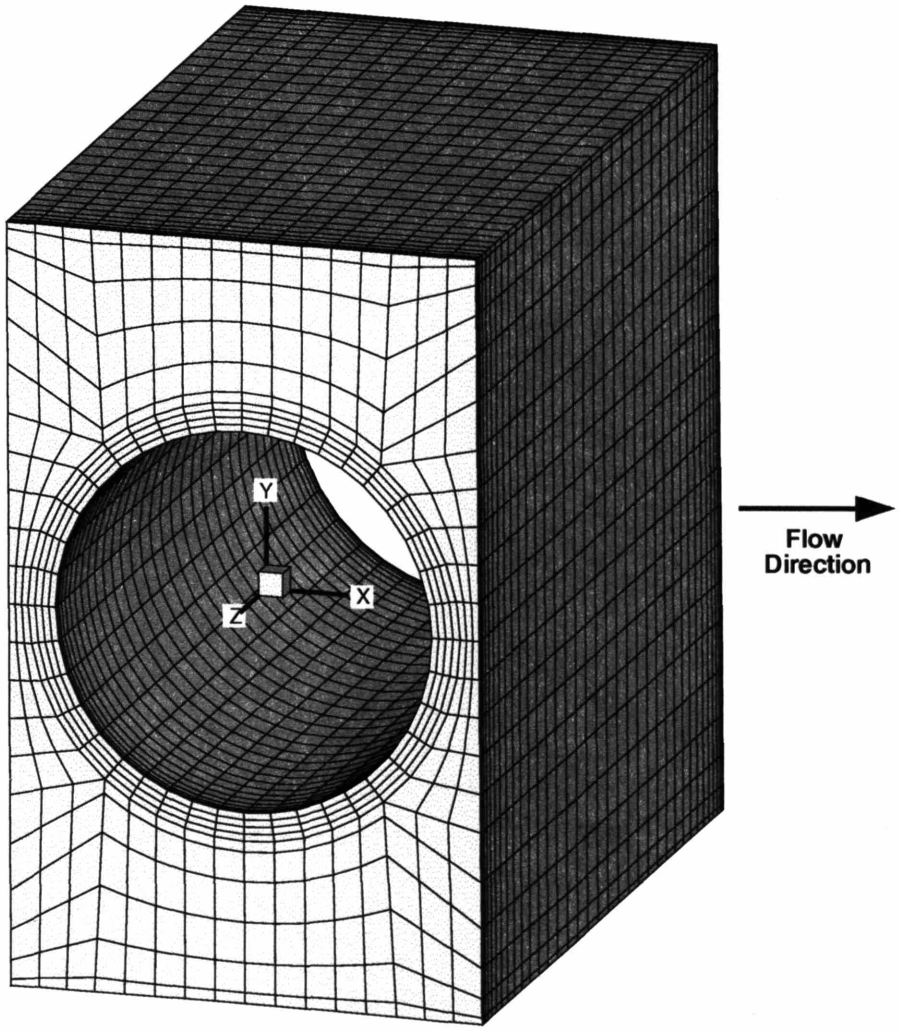


Figure 8-2: The mesh used for simulation of the three-dimensional bounded and periodic width geometries. The mesh is composed of 14420 hexahedral elements.

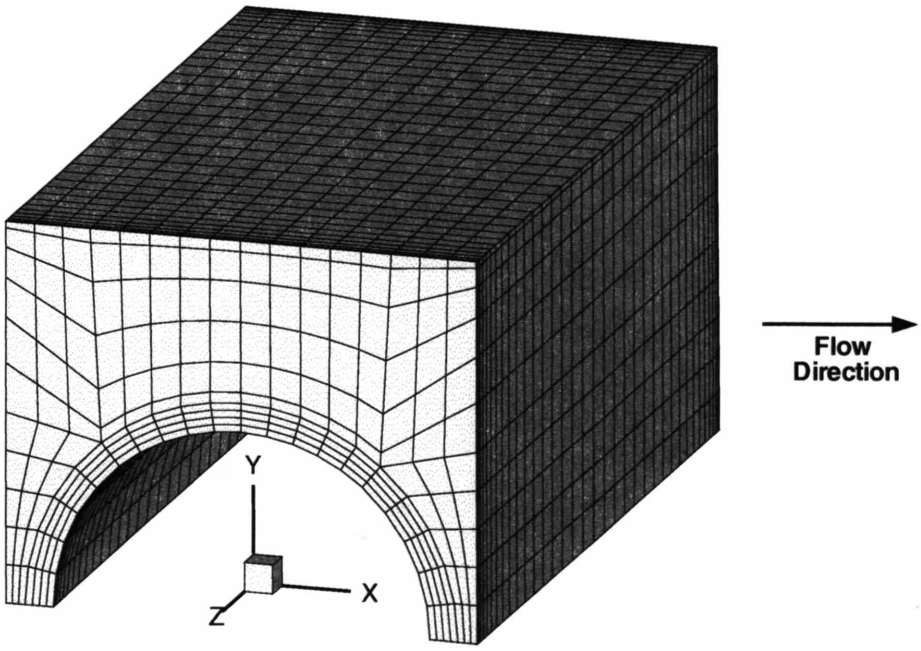


Figure 8-3: The symmetric mesh used for simulation of the three-dimensional bounded and periodic width geometries. The mesh is identical to the upper half of the three-dimensional mesh shown in Fig. 8-2 and is composed of 7210 hexahedral elements.

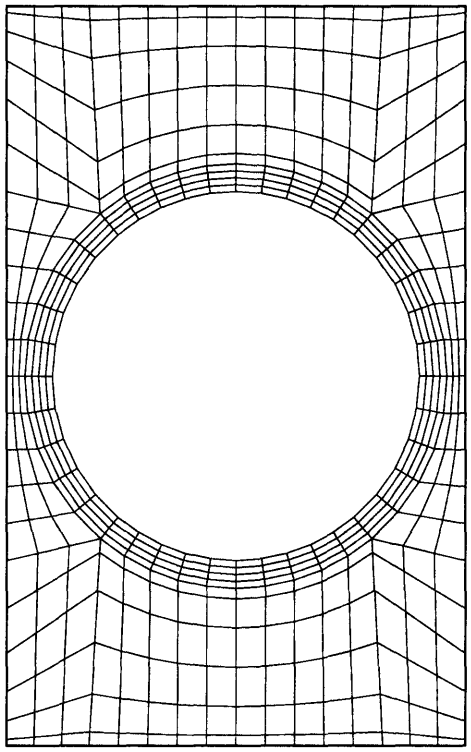


Figure 8-4: The two-dimensional mesh used in simulations of the periodic, linear array of cylinders where variations in the z direction have been neglected and the veloctiy in the z direction is assumed to be zero. The mesh is a cross-section of the three-dimensional mesh and consists of 412 quadrilateral elements.

8.2 Results

8.2.1 Newtonian fluid

Simulation results for the flow of a Newtonian fluid in the periodic, linear array of cylinders are presented here. Four different geometrical cases are modeled, including an array of infinite width where variations in the z direction and velocity in the z direction are neglected, an array of infinite width modeled where the flow is assumed to be periodic in the z direction with a period of $4R_c$, an array of infinite width modeled where the flow is assumed to be periodic in the z direction with a period of $4R_c$ and symmetric about the $y = 0$ plane, and an array bounded by solid walls in the z direction with a width of $4R_c$. Comparisons of the flows in the four geometries are also given.

Array of infinite width

Simulation results are given here for the flow of a Newtonian fluid in an array of infinite width where variations in the z direction and velocity in the z direction are neglected. The z direction is therefore referred to as the "neutral" direction and the model is reduced to two dimensions. Contour plots of the periodic portion of the pressure and the x and y components of velocity are shown in Fig. 8-5. Streamlines for the flow are constructed by first combining the components of velocity from the simulation to form a vector field over the entire flow domain. Streamtraces are then found much like experimental streakline images are constructed, by tracing the path of marker particles flowing through the domain. The streamtraces of the flow are equivalent to the streamlines at steady state. The streamlines for this flow are shown in Fig. 8-6. A pair of vortices fill the gap between the cylinders, each reaching outward $0.38R_c$ units from the centerline. The vortices act to divide the flow into four separate regions, namely the main from above the cylinders, the two vortices between the cylinders, and the main flow below the cylinders. Because of the closed loops of streamlines in the vortices, no information is exchanged between these regions.

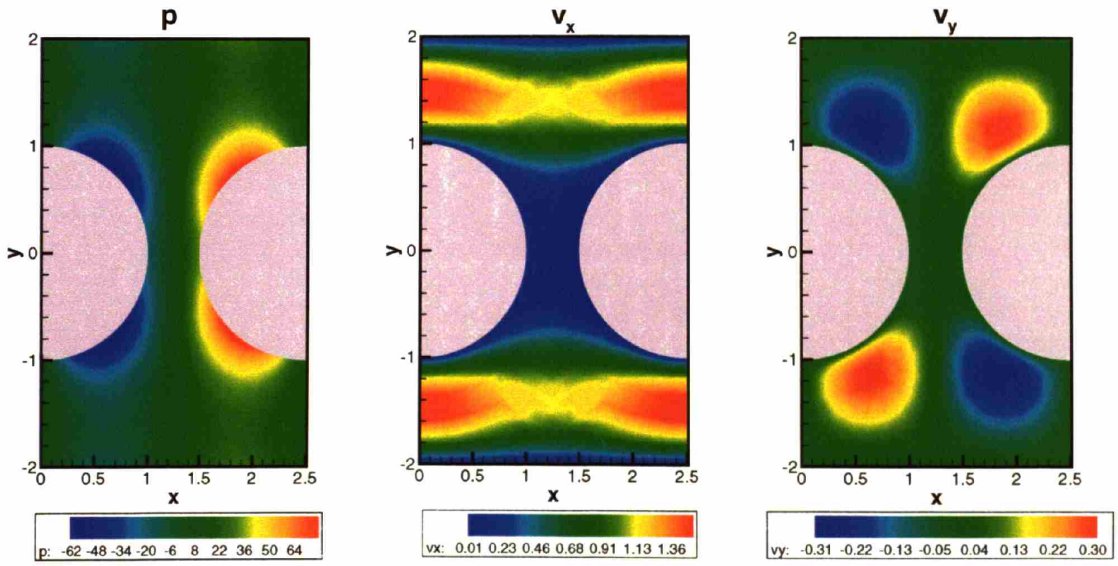


Figure 8-5: Contour plots of the periodic portion of pressure and the x and y components of the velocity for the flow of a Newtonian fluid in a periodic, linear array of cylinders with infinite width. Variations in the z direction and the z velocity are neglected.

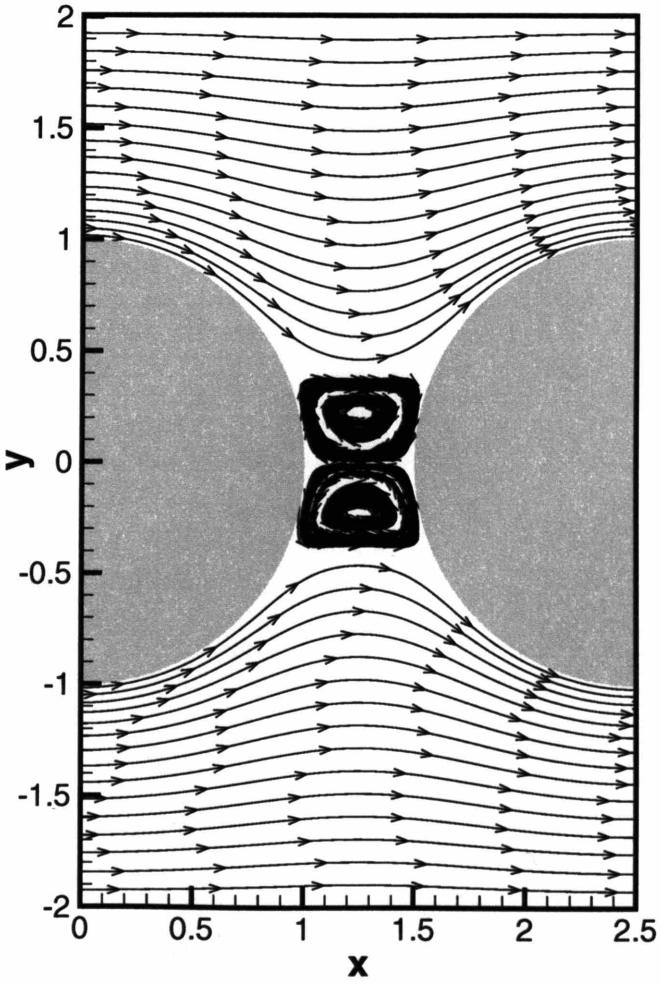


Figure 8-6: The streamlines for the flow of a Newtonian fluid in a periodic, linear array of cylinders with infinite width. Variations in the z direction and the z component of velocity are neglected.

Effects of considering variations in the z direction

Simulations were also performed using the three-dimensional geometry of width $4R_c$ where the flow is assumed to be periodic in the z-direction, and variations in the "neutral" direction and non-vanishing v_z are admitted. The flow field and pressure field from the simulations showed no variation in the "neutral" direction. Contours of the periodic portion of the pressure and the x and y velocity in the $z = 0$ plane are given in Fig. 8-7. The streamlines for this flow are presented in Fig. 8-8. The flow field and pressure

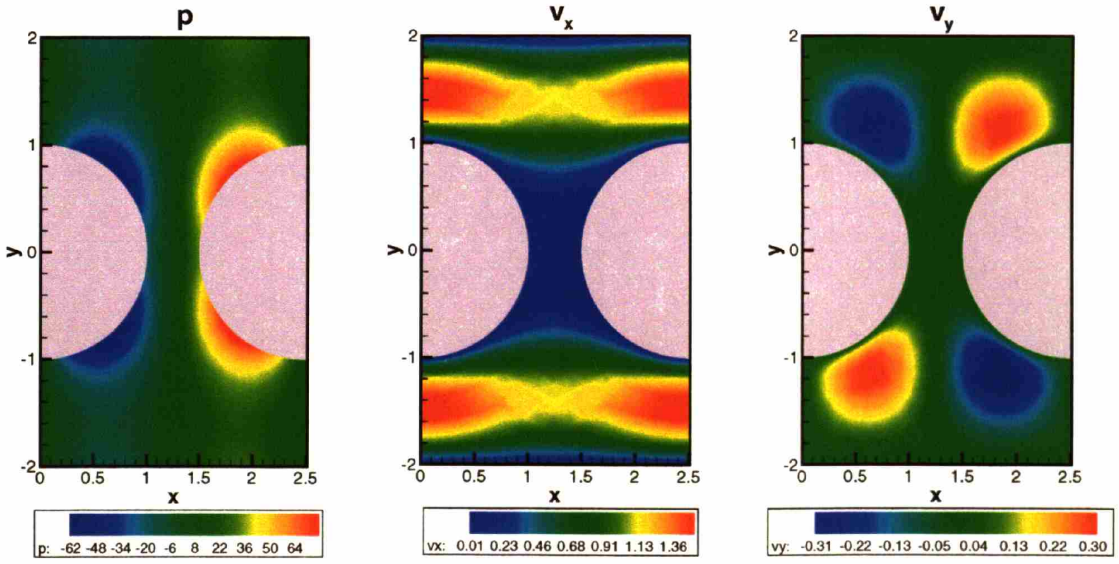


Figure 8-7: Contour plots of the periodic portion of pressure and the x and y components of the velocity in the $z = 0$ plane for the flow of a Newtonian fluid in a periodic, linear array of cylinders with periodic width of $W = 4R_c$.

field show virtually no deviation from the case with the assumed neutral z direction. Results of the simulations for this geometry with the added assumption that the flow is symmetric about the $y = 0$ plane also showed virtually no deviation from the case with the assumed neutral z direction.

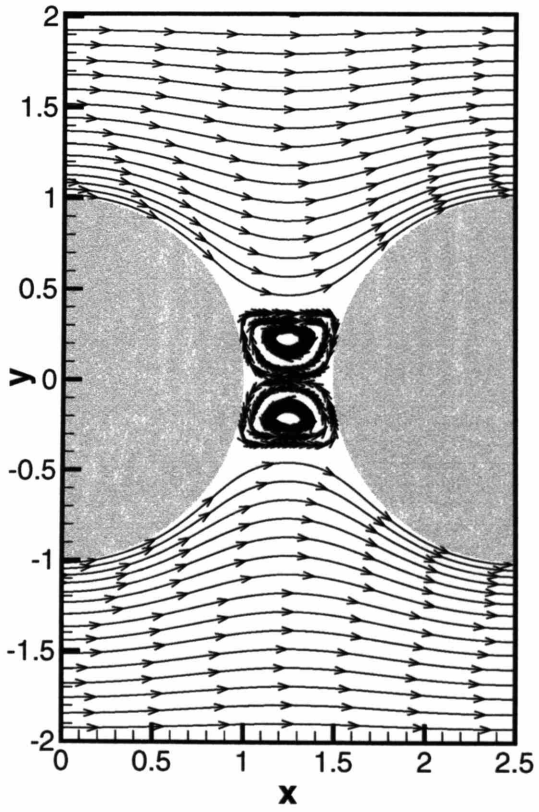


Figure 8-8: The streamlines of the flow of a Newtonian fluid through a periodic array of cylinders unbounded in the z direction with periodic width of $W = 4$. Flow is in the positive x direction.

Effects of adding bounding walls

Simulations of the periodic, linear array of cylinders with no slip walls bounding the flow in the z-direction and with width of $4R_c$ apart were performed to discern the effects of the sidewalls on the flow. While the flow field showed virtually no deviation from the case of assumed neutral z-direction when viewed at the centerplane $z = 0$, significant effects are seen in the $z = \pm R_c$ planes, the halfway points between the $z = 0$ plane and the sidewalls. Contours of the periodic portion of the pressure and the x and y components of velocity in the $z = R_c$ plane are given in Fig. 8-9. While the pressure field shows little variation, the bounding walls have acted to reduce the intensity of the flow in the x and y directions at a distance of R_c from the walls. The streamlines for this flow in the $z = 0$ and $z = R_c$ planes are presented in Fig. 8-10. Here the vortices in the gap between the cylinders are shown to shrink significantly at a distance of R_c from the bounding wall.

Figure 8-11 shows a close-up of the streamlines in the gap between the cylinders on the $z = 0$ plane, superimposed onto the contour field of the velocity in the y direction. Only two levels are used to discretize the contour levels, one greater than zero and one less than zero, allowing for easy identification of where the y velocity is zero. Near the cylinder surface the y velocity changes signs at the separation lines between the vortex and the main flow in the positive and negative half of the geometry as well as at the vortex-vortex interface at the $y = 0$ centerline. Extracting a line of data from the flow at $x = 1$ ($z = 0$) makes this fact readily apparent, as is shown in Fig. 8-12. Here the three intersections of the data and the dashed line denote the three sign changes. Comparison of v_y at $x = 1$ ($z = 0$) from the four simulation cases demonstrates only slight variation in this quantity, as shown in Fig. 8-13. Only slight differences in the separation line locations exist among the four different test cases.

To help identify the effects of adding bounding walls to the geometry, comparison of the y velocity near the aft edge of the cylinder is made closer to the bounding surfaces of the solid wall and the periodic wall cases. An isosurface plot of the $v_y = 0$ surface

near the aft edge of the cylinder is shown for the periodic wall case in Fig. 8-14 and for the solid wall case in Fig. 8-15. The isosurface is shown for the flow from the cylinder wall to plane halfway between the cylinders. The dashed red line indicates where the isosurfaces intersect the $x = 1$ plane. The periodic wall case shows no variation in the separation lines along the width of the domain. For the solid wall case, while the interface between the vortices, denoted by the central isosurface, exhibits virtually no variation over the width of the domain, the separation lines between the primary flow and the vortices shows significant variation across the width of the domain. Moving away from the $z = 0$ plane, the point of attachment of the vortices decreases in the y direction. The bounding walls clearly impose a significant effect on the flow over nearly the entire width of the domain.

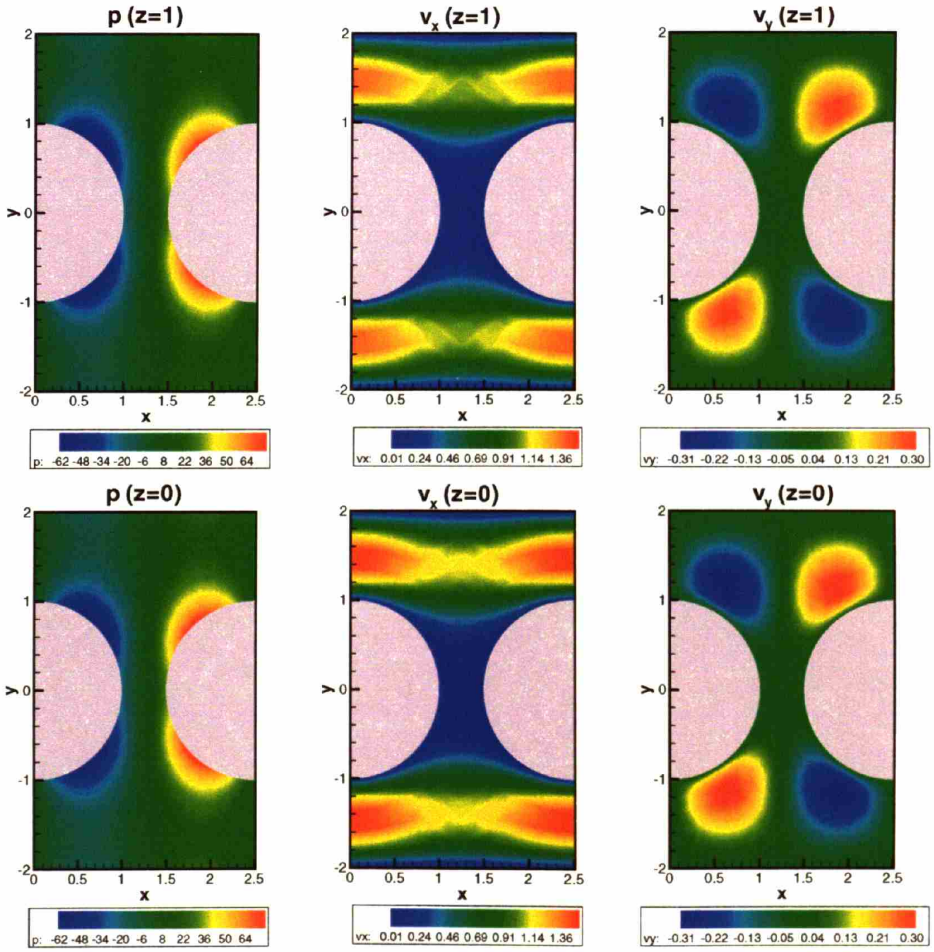


Figure 8-9: Contour plots of the periodic portion of pressure and the x and y components of the velocity in the $z = 1$ plane for the flow of a Newtonian fluid in a periodic, linear array of cylinders of width $W = 4R_c$.

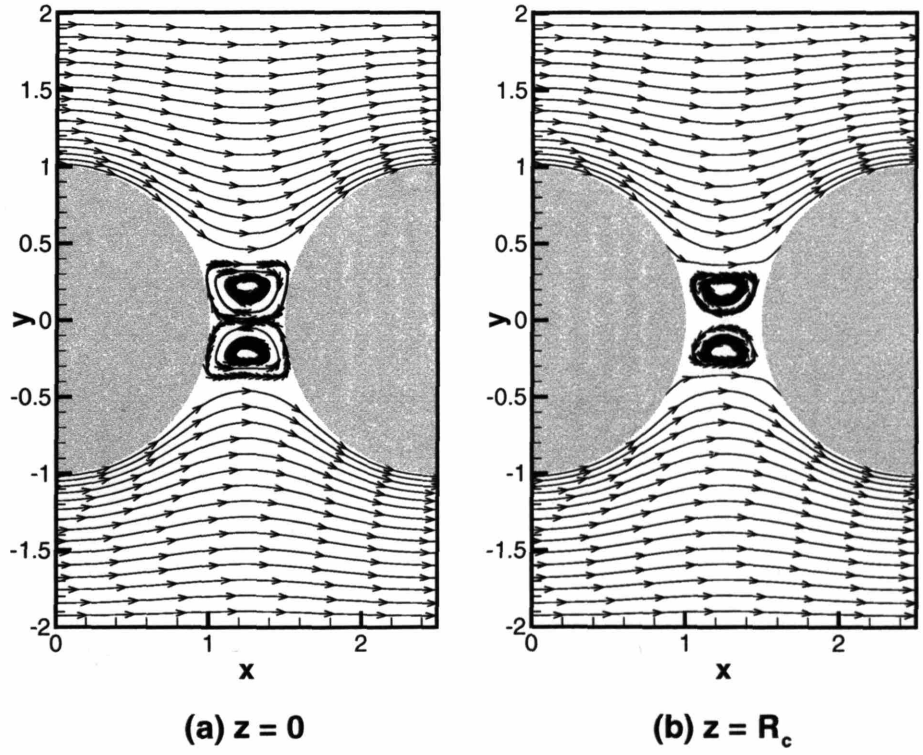


Figure 8-10: The streamlines of the flow of a Newtonian fluid through a periodic array of cylinders bounded in the z direction with width $W = 4R_c$. (a) $z = 0$ plane; (b) $z = R_c$ plane. Flow is in the positive x direction.

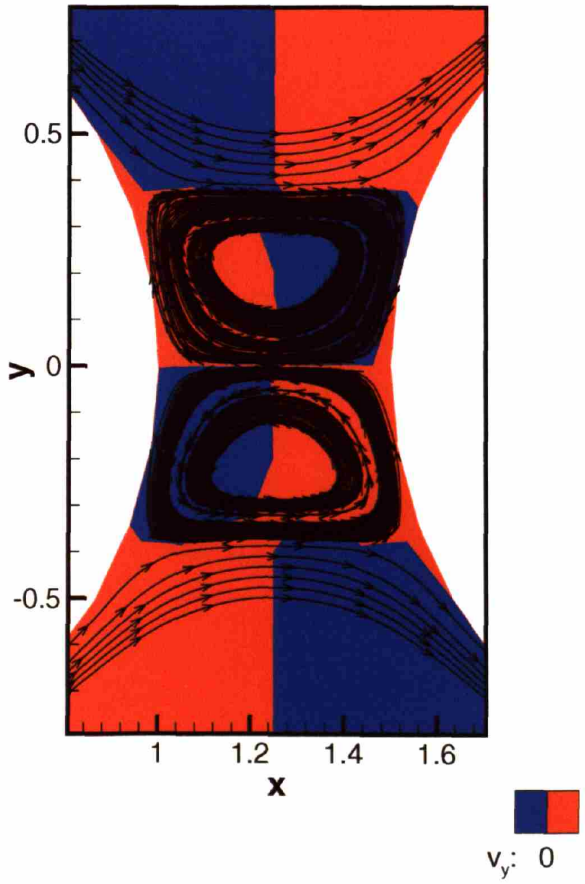


Figure 8-11: Plot of streamlines near the gap between the cylinders overlaid on a contour plot of the positive and negative regions of the y component of velocity.

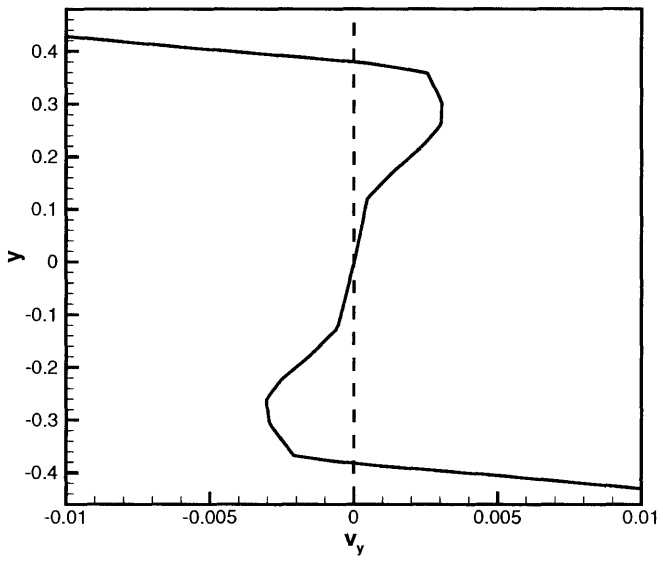


Figure 8-12: Velocity in the y direction along the line $x = 1$ ($z = 0$). The intersections with the $v_y = 0$ line correspond to the separation lines between the main flow and the vortex, and the vortex/vortex interface.

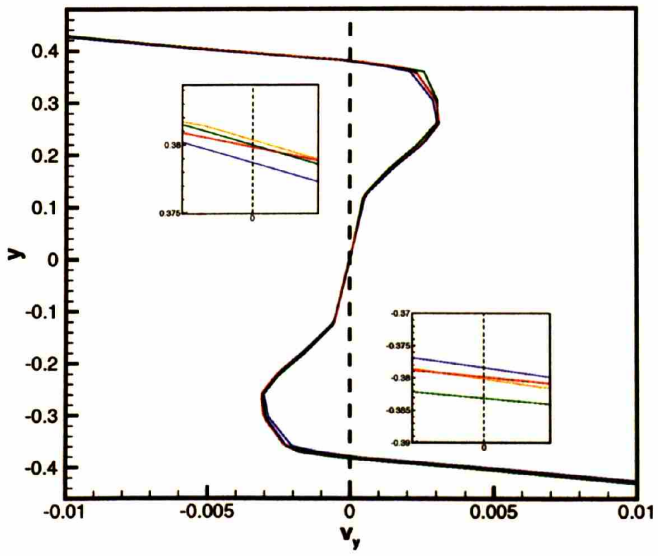


Figure 8-13: Velocity in the y direction along the line $x = 1$ ($z = 0$) for the four variations in the width of the periodic, linear array of cylinders. (green): infinite width, no z variations; (orange): periodic width; (red): periodic width, forced symmetry; (blue): bounded width.

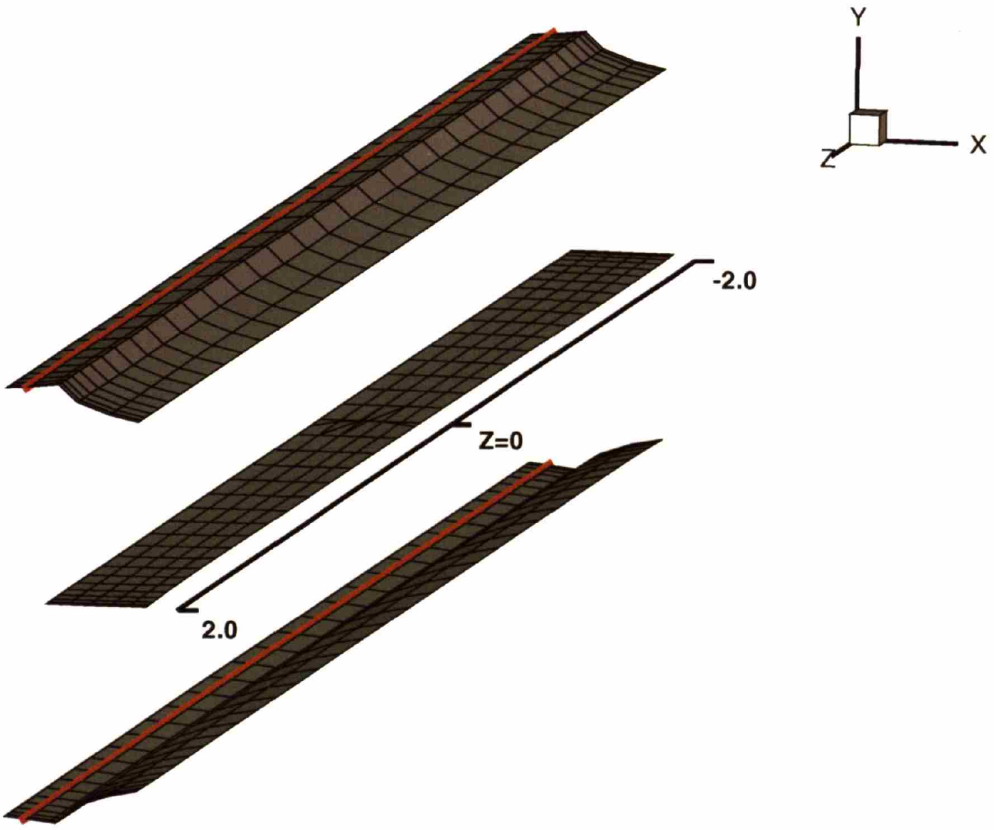


Figure 8-14: Isosurface for $v_y = 0$ near the downstream edge of the cylinder for a Newonian fluid flowing in the periodic, linear array of cylinders with periodic width of $W = 4R_c$. The region between the downstream cylinder wall and the yz plane at $x = 1.25$ is shown. The upper and lower isosurface near the cylinder wall denotes the interface between the main flow and the vortices. The center isosurface denotes the interface between the vortices. The red line is at $x = 1$.

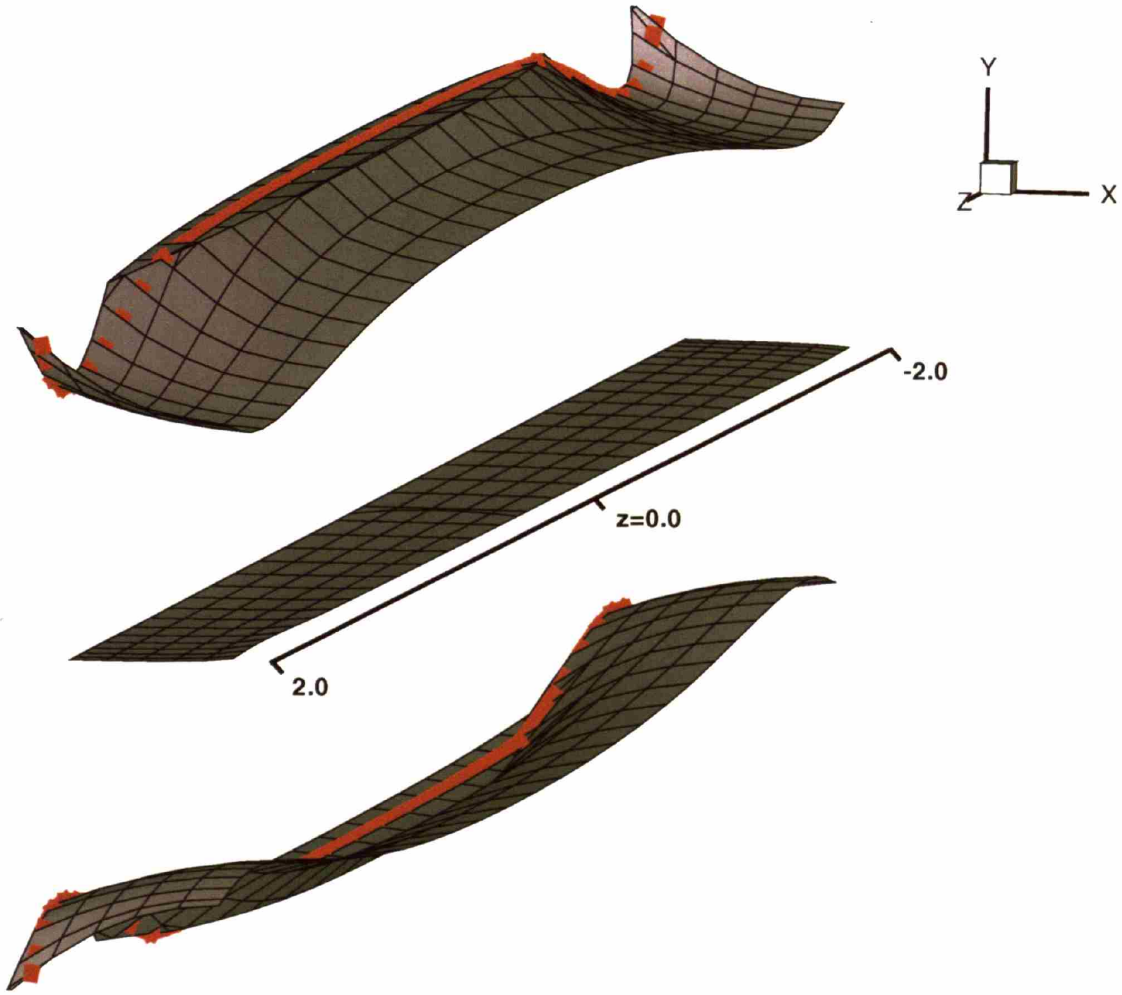


Figure 8-15: Isosurface for $v_y = 0$ near the downstream edge of the cylinder for a Newontian fluid flowing in the periodic, linear array of cylinders with periodic width of $W = 4R_c$. The region between the downstream cylinder wall and the yz plane at $x = 1.25$ is shown. The upper and lower isosurface near the cylinder wall denotes the interface between the main flow and the vortices. The center isosurface denotes the interface between the vortices. The red line is at $x = 1$.

8.2.2 Oldroyd-B model fluid

This section explores the effects of increasing viscoelasticity on the flow. Results from simulations of the flow of an Oldroyd-B fluid in the periodic, linear array of cylinders are presented. As with the Newtonian fluid case, four different geometrical cases are modeled, namely an array of infinite width where variations in the z direction and velocity in the z direction are neglected, an array of infinite width modeled where the flow is assumed to be periodic in the z direction with a period of $4R_c$, an array of infinite width modeled where the flow is assumed to be periodic in the z direction with a period of $4R_c$ and symmetric about the $y = 0$ plane, and an array bounded in the z direction with a width of $4R_c$. Comparison of the flows in the four geometries is given along with comparison to trends found in previous works.

Array of infinite width

Simulation results are given here for the flow of an Oldroyd-B fluid with $\beta = 0.67$ in an array of infinite width where variations in the z direction and velocity in the z direction are neglected. The streamlines in the gap between the cylinders are shown for a range of De in Fig. 8-16. The simulations show that increasing the viscoelasticity of the flow causes the vortices in the gap between the cylinders to grow outward from the $y = 0$ plane. This appears to be caused by an increased resistance of the fluid to curvature along the streamlines as the fluid passes over the gap between the cylinders. The vortices also show increased concavity as De is increased, with more growth on the edge of the downstream cylinder than the upstream cylinder. For comparison, results reproduced from [73] showing a single streamline representing the outline of the upper vortex computed at De = 1.46 is shown in Fig. 8-17. The outline of the upper vortex computed by Smith et al. compares favorably with the current simulation results, especially considering the relatively low resolution in the xy plane of the mesh used in the present work. Steady state solutions were not found for De higher than 1.5. Simulations at De of 1.6 and higher exhibited oscillations with increasing amplitude in the L_2 norms of both $\boldsymbol{\tau}_p$ and $\mathbf{v}$.

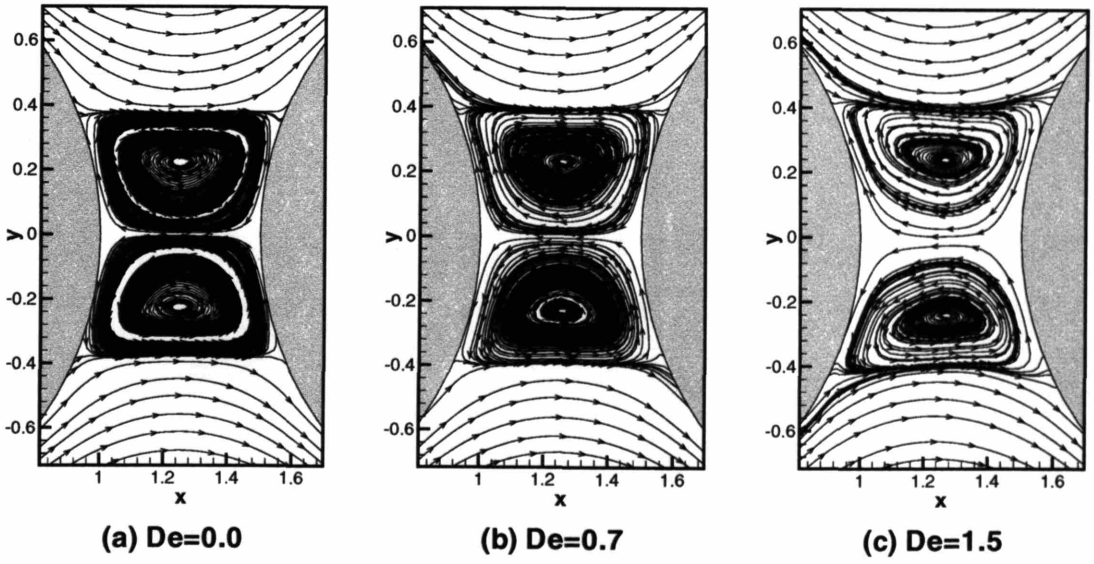


Figure 8-16: Streamlines computed for the flow of an Oldroyd-B fluid with $\beta = 0.67$ and De of 0.0, 0.7, and 1.5 in a periodic, linear array of cylinders with infinite width where variations in the z direction are neglected and v_z is assumed to be zero.

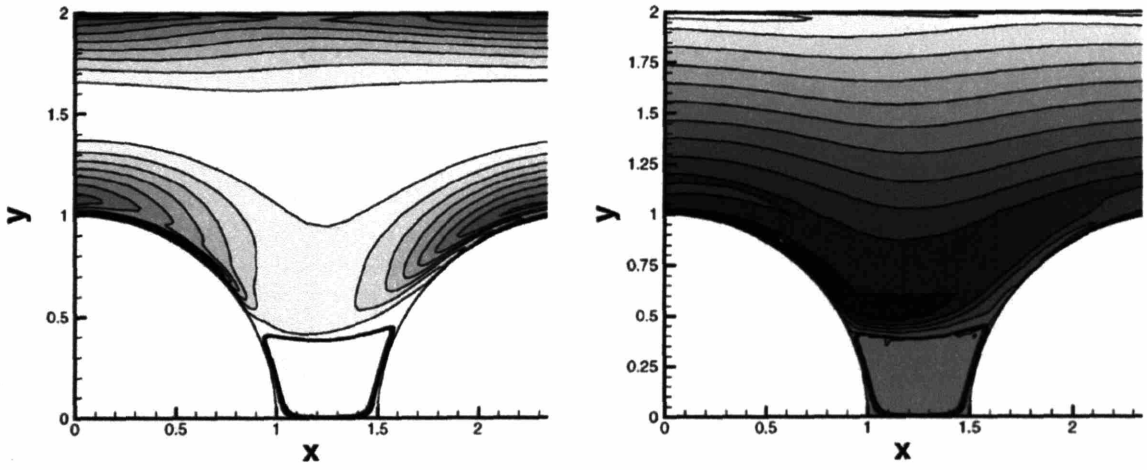


Figure 8-17: Contours of τ_{p11} (left) and τ_{p12} (right) at We=1.46 as computed by Smith et al. [73] where 1 and 2 denote the streamwise tangential and streamwise normal direction in the Protean coordinate system. The dark outline in the gap between the cylinders is the computed vortex.

Oscillations were also witnessed in the L_2 norms of the De = 1.5 case, but the amplitude of the oscillations decreased with time, eventually reaching zero. This limiting value of De is within 2% of the critical value found by Smith et al.. Care was taken to ensure that the simulations were converged in the timestep demonstrating that the oscillations were not due to a lack of temporal resolution.

Using the analysis developed above for the Newtonian cases, the relative change in vortex size with increasing De can be shown by viewing the location of the sign change in v_y in the $x = 1$ plane, Fig. 8-18. The flow remains stable and symmetric for De ≤ 1.5.

The drag on the surface of a single cylinder in the periodic array as a function of De is computed for comparison to previous work. The drag force $\mathbf{F}_D$ of a fluid on a surface is calculated from the expression

$$\mathbf{F}_D = -\int_S (\boldsymbol{\delta}_r \cdot \{p\boldsymbol{\delta} + \boldsymbol{\tau}\})\, dS = -\int_S (\boldsymbol{\delta}_r \cdot \{p\boldsymbol{\delta} + \boldsymbol{\tau}_p - \beta\dot{\boldsymbol{\gamma}}\})\, dS \tag{8.5}$$

where $\boldsymbol{\delta}_r$ is an outward unit vector normal to the surface. The drag force is non-

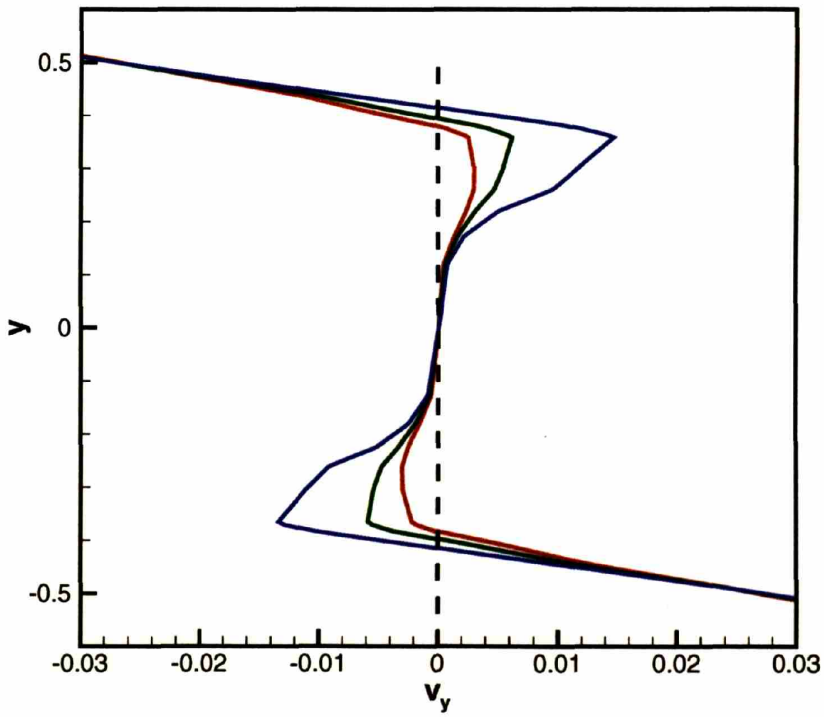


Figure 8-18: Velocity in the y direction along the line $x = 1$ for the flow of an Oldroyd-B fluid of $\beta = 0.67$ in a periodic, linear array of cylinders with infinite width and no variation in the z direction. (red): De=0.0; (green): De=0.7; (blue): De=1.5.

dimensionalized with the characteristic viscous stress, $\eta_0 \langle v \rangle / R_c$. The drag force computed on the cylinder in the two-dimensional simulations is shown in Fig. 8-19. Here, $\chi = \mathbf{F}_D / \mathbf{F}_{D,N}$ is the drag force scaled by the drag force of a Newtonian fluid. Liu et al. [54] reported that the drag force of an Oldroyd-B fluid on a single cylinder in the periodic, linear array of cylinders with $L_c = 2.5$ and $H = 2$ reaches a minimum at some critical value of De and then grows monotonically for higher De. The drag force computed in the two-dimensional simulations shows the same trend with increasing De, but the values do not agree quantitatively with the results of Liu et al..

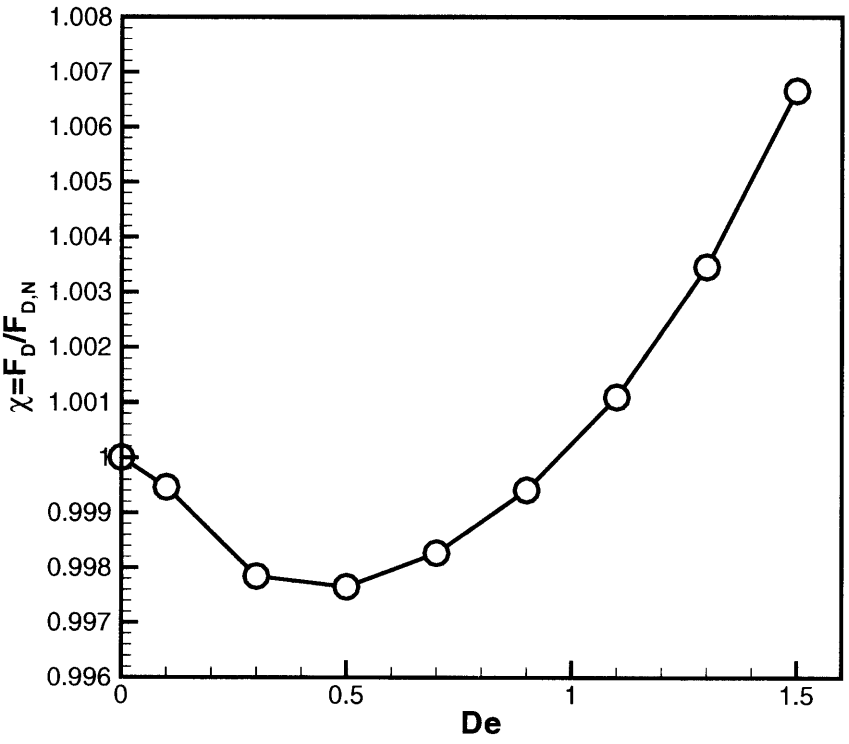


Figure 8-19: The normalized drag force on a single cylinder with increaseing Deborah number for the flow of an Oldroyd-B fluid of $\beta = 0.67$ through a periodic, linear array of cylinders with $L_c = 2.5$.

Effects of considering variations in the z direction

To model the effects of including variations in the z direction, simulations for the geometry with periodic width of $W = 4R_c$ were computed up to De = 0.725. At De leading up to 0.725, increased sensitivity to the size of the timestep was found. Whereas the simulations at De of 0.1, 0.3, and 0.5 all were stable at a dimensionless timestep of no greater than $\Delta t = 0.01 * \langle v \rangle / h$, where $\langle v \rangle / h$ is the characteristic timescale of the flow, the De=0.7 case was time-stable at $\Delta t = 0.0035$. Figure 8-20 shows the stable timestep as a function of De for the stable cases. Error bars are included that show the half-interval

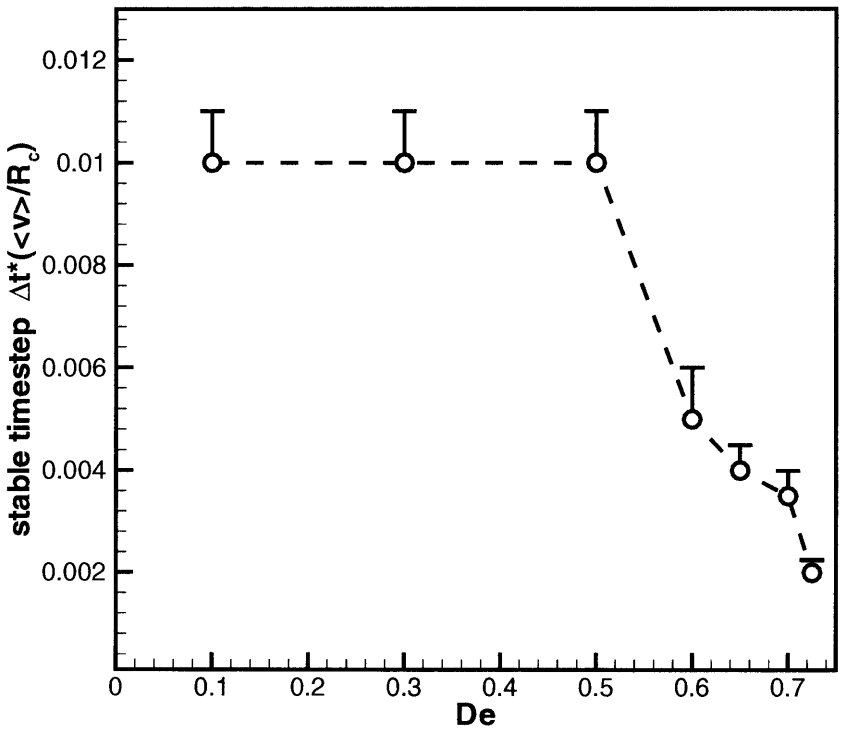


Figure 8-20: Stable dimensionless timestep as a function of Deborah number for the simulations of an Oldroyd-B fluid with $\beta = 0.67$ in a periodic, linear array of cylinders with periodic width of $W = 4R_c$. Error bars are given that show the half-interval over which the timestep was tested for temporal stability in the simulation at each value of De.

over which timesteps were chosen to test for stability in the simulation at each De. Only the top half of the error bar is included indicating that the simulation is known to be stable for timesteps smaller than the data point. This suggests that another timescale in addition to the process timescale has appeared and is affecting the numerical stability of the governing equations. At higher De the simulations failed to reach a stready-state solution and exhibited growing oscillations in the L_2 norms of $\boldsymbol{\tau}_p$ and $\mathbf{v}$. Care was taken to ensure that the simulations were converged in the timestep.

The vortices formed along the $z = 0$ plane with increasing De are shown in Fig 8-21. The vortices are very similar to those found in the two-dimensional simulation up to

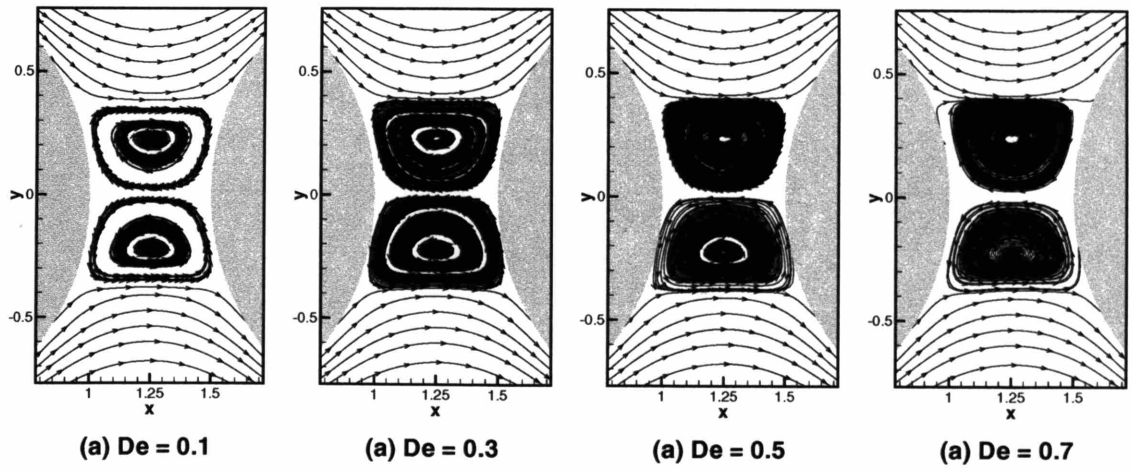


Figure 8-21: Streamlines computed for the flow of an Oldroyd-B fluid with $\beta = 0.67$ and De of 0.1, 0.3, 0.5, and 0.7 in a periodic, linear array of cylinders with a periodic width of $W = 4R_c$.

De = 0.7. No evidence of the concave shape of the outer edges of the vortices that is just starting to appear in the two-dimensional simulations at De = 0.7 is present here, though similar growth of the vortices with increasing De is apparent.

Plots of the xx, yx, and yy components of $\boldsymbol{\tau}_p$ in the $z = 0$ plane are shown in Figs. 8-22, 8-23, and 8-24, respectively. Buildup of stress with increasing De is mostly localized near the cylinder surface between 60 and 90 degrees above and below the centerline of

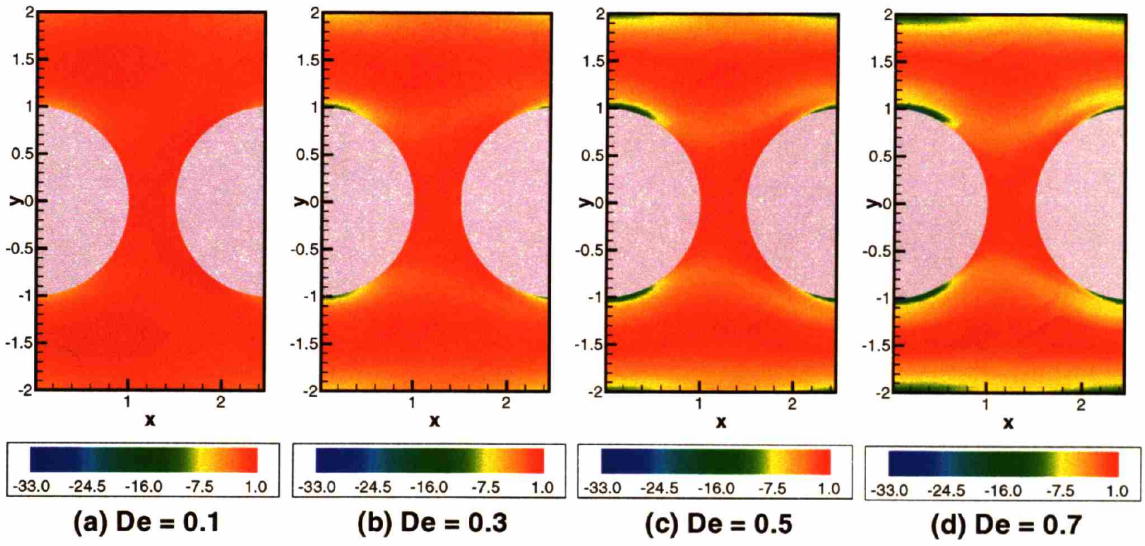


Figure 8-22: Contour plot of the xx component of $\boldsymbol{\tau}_p$ for the flow of an Oldroyd-B fluid with $\beta = 0.67$ for a range of De in the periodic, linear array of cylinders with periodic width of $W = 4R_c$. Flow is in the positive x direction.

the geometry. Also, a buildup of $\boldsymbol{\tau}_{pxx}$ with De is seen at the top and bottom walls of the geometry. Furthermore, as De increases, a break in the symmetry of the stress field becomes apparent, most prevalent in $\boldsymbol{\tau}_{pyy}$ where the lobe growing off the bottom upstream side of the cylinder is clearly more intense than that on the top upstream side for De of 0.5 and 0.7. Stress fields for the xz, yz, and zz components of $\boldsymbol{\tau}_p$ were essentially zero for all De simulated.

The only significant variations in the z direction are found in the z-component of velocity, though this component's values are quite small compared to v_x, and v_y. Contour plots of the v_z velocity in the $y = \pm 1.5$ planes are shown in Fig. 8-25. Recall that the cylinder axes are at $x = 0$ and $x = 2.5$, with the gap present at $1 < x < 1.5$. Though only very weak flow is present in the z direction, the variation is clearly present. On the downstream side of the cylinder, the flow is shown to move in the positive z direction while on the upstream side of the cylinder the flow moves in the negative z direction. This suggests that large recirculation cells are present in the flow traveling over the cylinders.

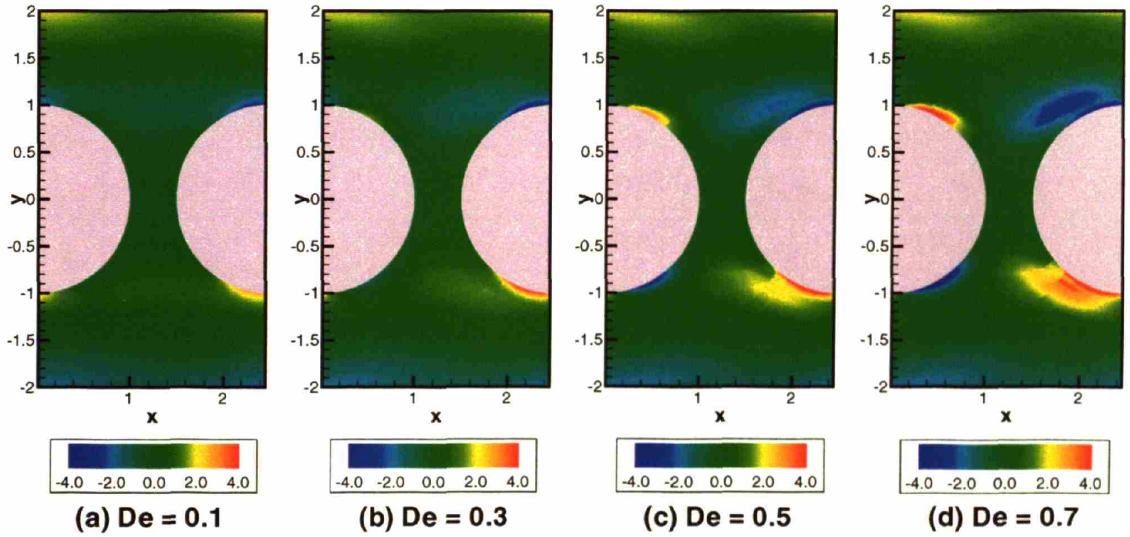


Figure 8-23: Contour plot of the yx component of $\boldsymbol{\tau}_p$ for the flow of an Oldroyd-B fluid with $\beta = 0.67$ for a range of De in the periodic, linear array of cylinders with periodic width of $W = 4R_c$. Flow is in the positive x direction.

While this is true above and below the cylinders, v_z is clearly not symmetric about the $y = 0$ plane. Above the cylinders the flow in the z direction on the downstream side of the cylinder is more concentrated in the center of the domain, while on the upstream side of the cylinder it is more concentrated near the outer edges of the domain. v_z below the cylinders is also considerably higher in magnitude than above the cylinders. As De is increased, the difference of the flow in the z direction on the upstream and downstream sides of the cylinder for $y = 1.5$ becomes more dramatic, while for $y = -1.5$ just the magnitude of the flow increases.

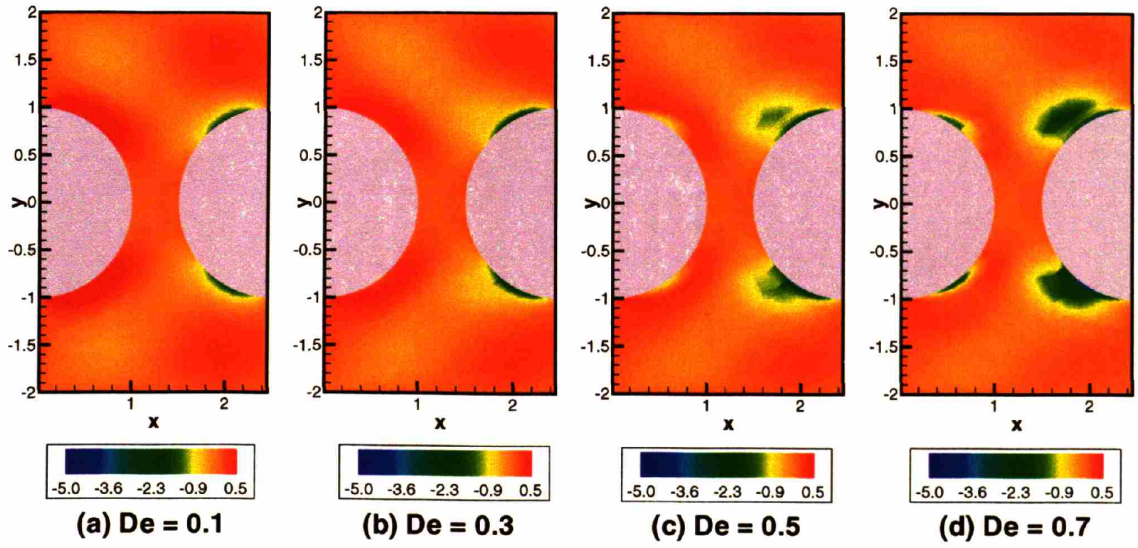


Figure 8-24: Contour plot of the yy component of $\boldsymbol{\tau}_p$ for the flow of an Oldroyd-B fluid with $\beta = 0.67$ for a range of De in the periodic, linear array of cylinders with periodic width of $W = 4R_c$. Flow is in the positive x direction.

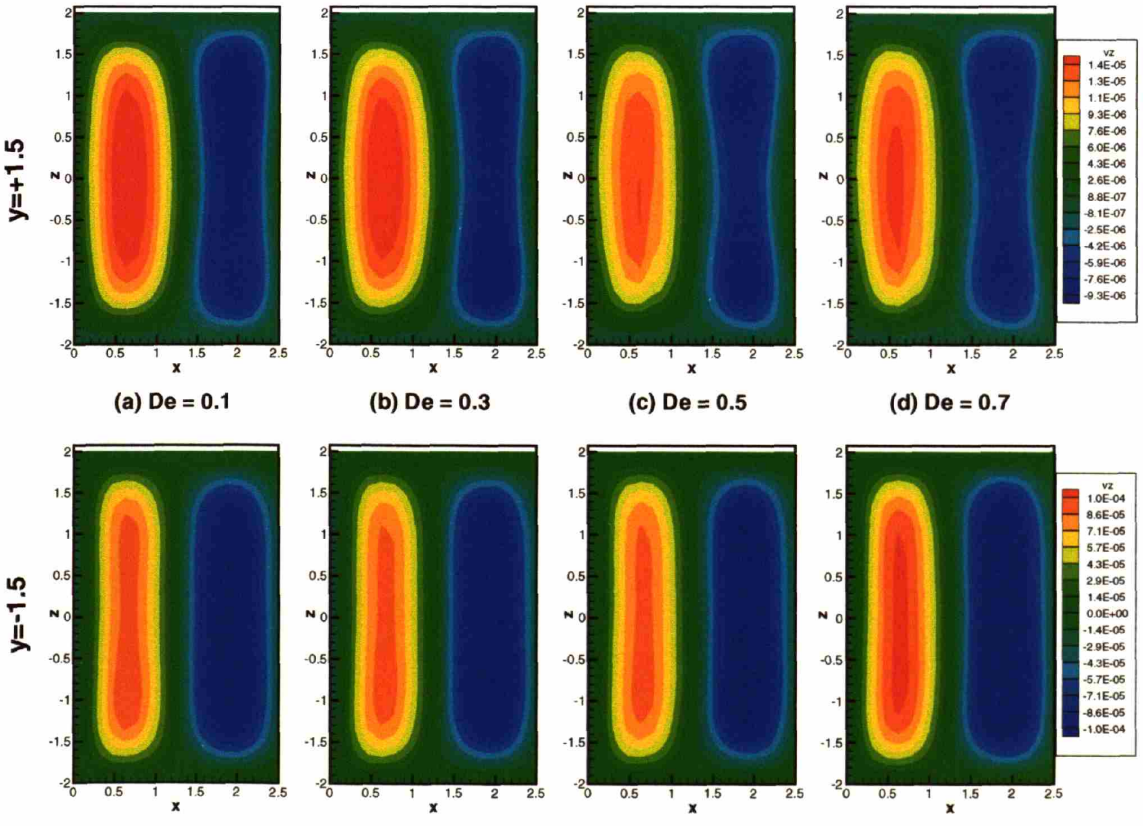


Figure 8-25: Contour plot of the v_z in the $y = 1.5$ and $y = -1.5$ planes for the flow of an Oldroyd-B fluid with $\beta = 0.67$ for a range of De in the periodic, linear array of cylinders with periodic width of $W = 4R_c$. Flow is in the positive x direction.

Effect of assuming symmetry about the $y = 0$ plane

Since the flow in the full geometry with periodic width of $W = 4R_c$ was found to be asymmetric, simulations of the same geometry with the added assumption that the flow field and stress fields are symmetric about the $y = 0$ plane were performed. As with the full simulation, increased sensitivity to the timestep was found as De was increased in the simulations, and the simulations failed to reach steady state above De of 0.725. Virtually no differences are present in the flow fields of the symmetric and the full simulation. This was anticipated, since symmetry in the flow field was preserved in all but the z direction in which it was considerably weaker than that in the x and y directions.

Plots of the xx, yx, and yy components of $\boldsymbol{\tau}_p$ in the $z = 0$ plane over the range of De simulated are shown in Figs. 8-26, 8-27, and 8-28. Comparison of the τ_{pyy} contours for

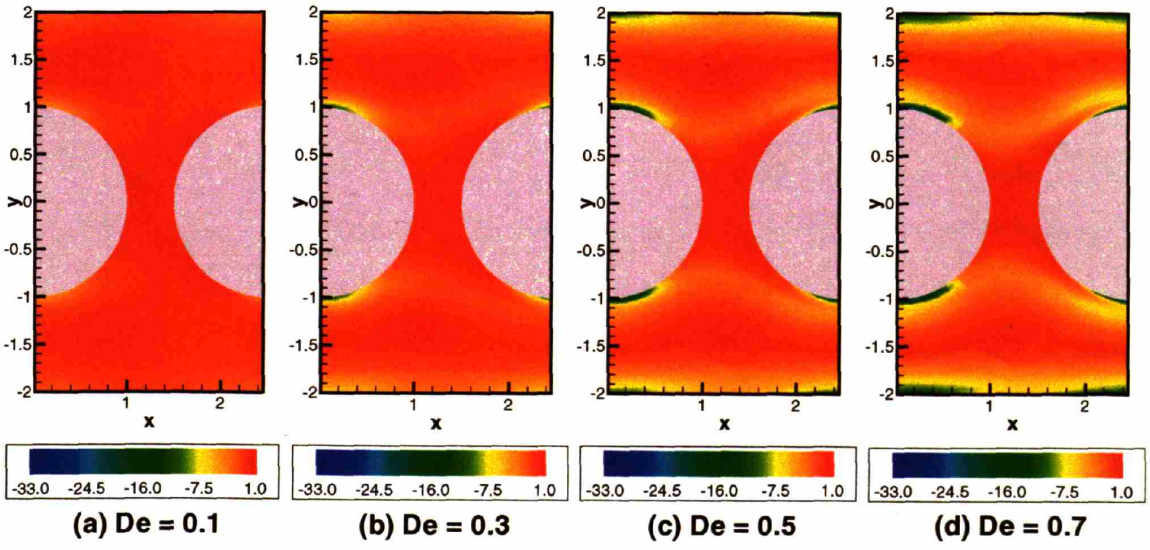


Figure 8-26: Contour plot of the xx component of $\boldsymbol{\tau}_p$ for the flow of an Oldroyd-B fluid with $\beta = 0.67$ for a range of De in the periodic, linear array of cylinders with periodic width of $W = 4R_c$. Symmetry is assumed on the $y = 0$ plane. Flow is in the positive x direction.

the symmetric case and full simulation where the break in symmetry was most apparent shows that the portion of the τ_{pyy} above the $y = 0$ plane in the full simulation more

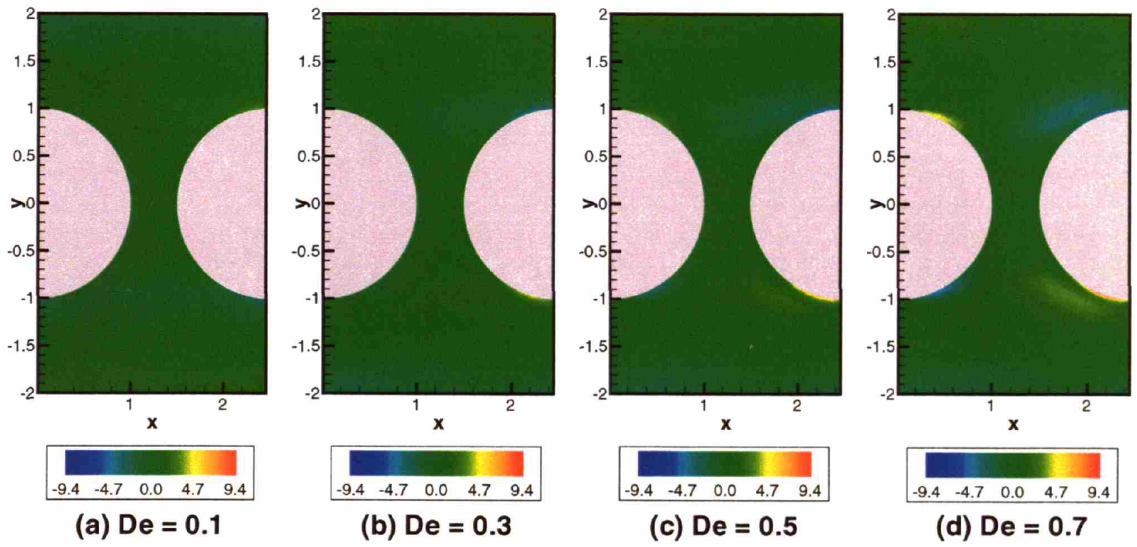


Figure 8-27: Contour plot of the yx component of $\boldsymbol{\tau}_p$ for the flow of an Oldroyd-B fluid with $\beta = 0.67$ for a range of De in the periodic, linear array of cylinders with periodic width of $W = 4R_c$. Symmetry is assumed on the $y = 0$ plane. Flow is in the positive x direction.

closely agrees with that of the symmetric simulation, while the portion below the $y = 0$ plane shows significantly higher values of stress near the back edge of the cylinder than the corresponding symmetric results. Differences are also apparent in the contours of τ_{pyx} where the shear stress is significantly in the full simulation for the same region where the differences in τ_{pyy} are present.

Plots of v_z in the $y = 1.5$ plane over the range of De simulated are given in Fig. 8-29. Like the components of the stress tensor, v_z in the symmetric simulations shows more similarity to the the upper portion of the solution from the full simulations than the lower portion where again on the downstream side of the cylinder v_z is larger in the center of the domain, while on the upstream side of the cylinder it is larger near the outer edges of the domain. This difference again becomes more dramatic with increasing De.

Since only half the number of elements are needed in the symmetric geometry to attain the level of resolution used in the full geometry, calculations with refined meshes

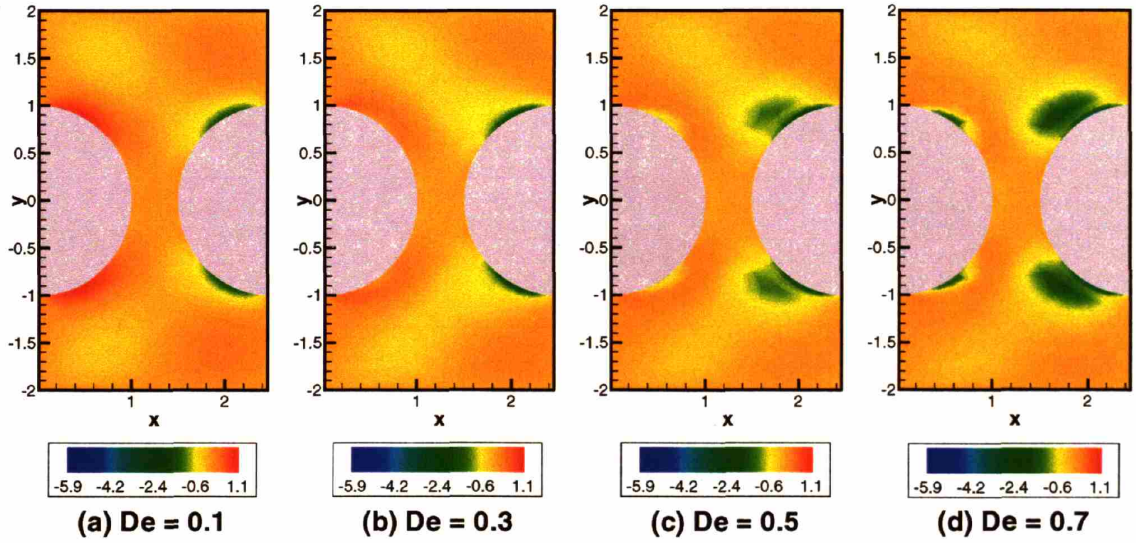


Figure 8-28: Contour plot of the yy component of $\boldsymbol{\tau}_p$ for the flow of an Oldroyd-B fluid with $\beta = 0.67$ for a range of De in the periodic, linear array of cylinders with periodic width of $W = 4R_c$. Symmetry is assumed on the $y = 0$ plane. Flow is in the positive x direction.

were possible. The goal of these calculations was to determine if it is possible to simulate the three-dimensional periodic cylinder flow with De greater than 0.7 by increasing mesh resolution. Two cases were considered: doubling the number of elements in the z direction, and doubling the number of elements in the xy plane. While doubling the number of elements in the z direction showed no difference in the maximum De that could be simulated, doubling the number of elements in the xy plane increased the maximum De at which steady state solution could be found to 0.9. No significant structural differences were seen in the flow field and stress fields at De of 0.9 for the resolved mesh when compared with those of the De of 0.7 case.

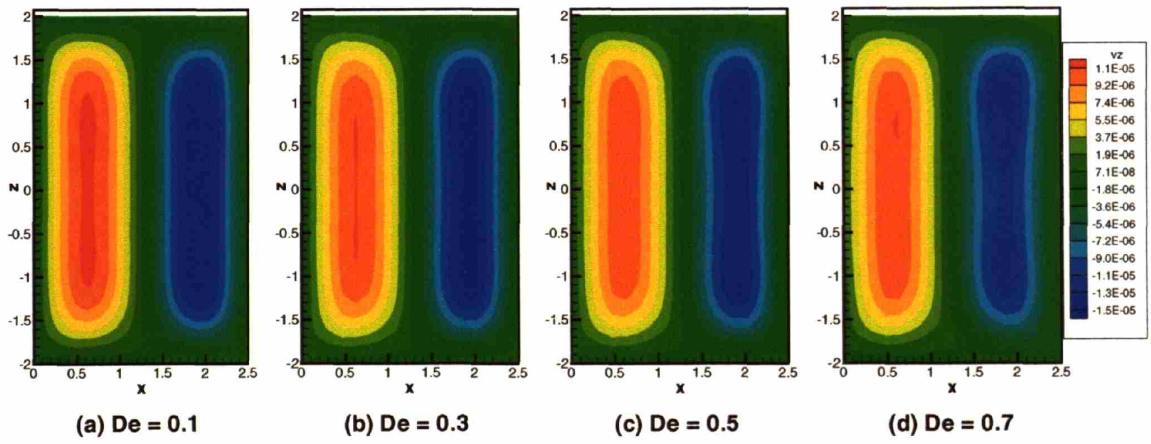


Figure 8-29: Contour plot of v_z in the $y = 1.5$ plane for the flow of an Oldroyd-B fluid with $\beta = 0.67$ for a range of De in the periodic, linear array of cylinders with periodic width of $W = 4R_c$. Symmetry is assumed on the $y = 0$ plane. Flow is in the positive x direction.

Effects of including bounding walls

To model the effects of bounding the flow in the z direction, simulations of the geometry of width $W = 4R_c$ were performed. As with the other three-dimensional cases, steady state solutions were only found up to De near 0.7. Comparisons were made to the unbounded cases on the $z = 0$ plane of the flow. The most noticeable differences appear in the streamlines of the flow with increasing De for the bounded case. Vortices formed along the $z = 0$ plane with increasing De are shown in Fig. 8-30. Unlike the flows in

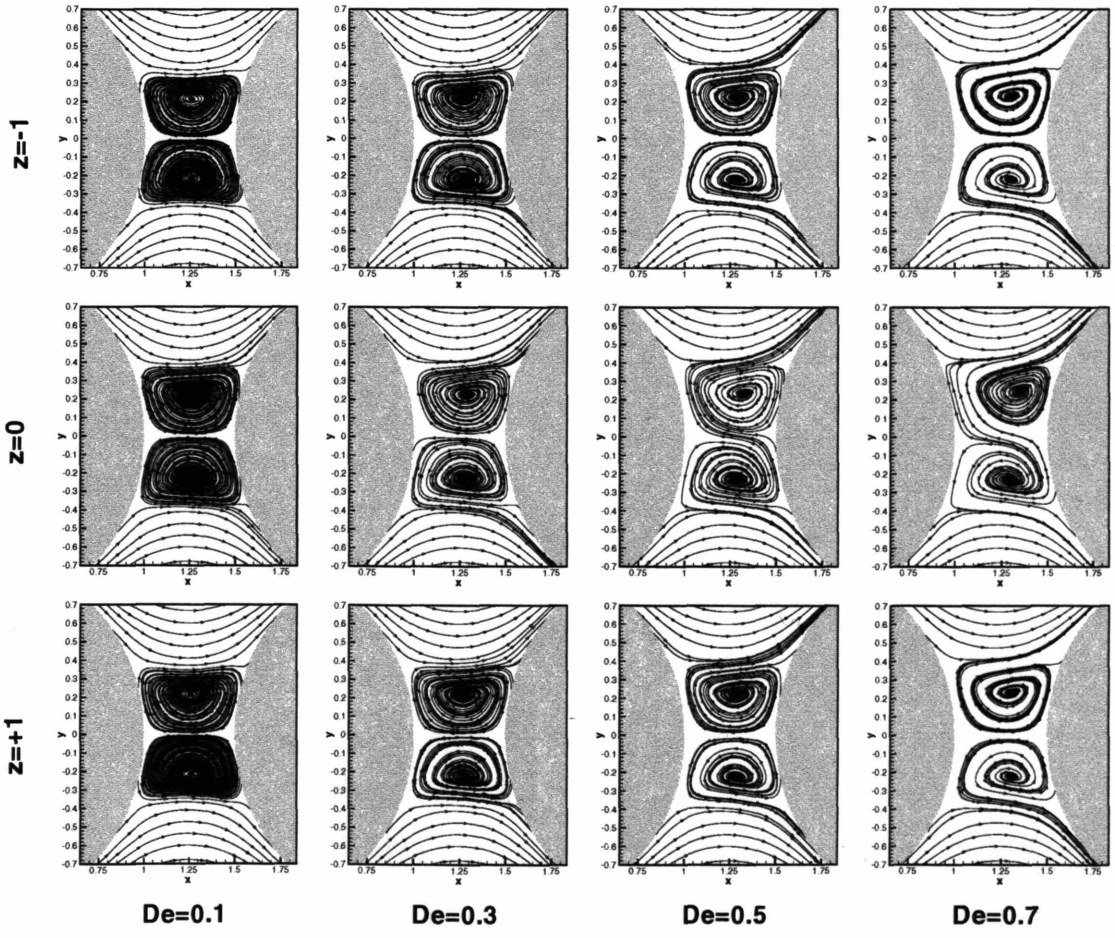


Figure 8-30: Streamlines in the $z = 0$ and $z = \pm 1$ planes computed for the flow of an Oldroyd-B fluid with $\beta = 0.67$ and De of 0.1, 0.3, 0.5, and 0.7 in a periodic, linear array of cylinders with width of $W = 4R_c$.

the infinite width and periodic width geometries, the vortices in the $z = 0$ plane deviate significantly from the Newtonian fluid with increasing De. At De $= 0.3$ the eye of the upper vortex begins to move forward and continues to do so up to De $= 0.7$, breaking the symmetry on the $y = 0$ plane. Flow across the $y = 0$ plane is also observed. The streamlines in the $z = 1$ and $z = -1$ planes do not show evidence of a break in symmetry. The eyes of both the upper and lower vortex move forward in the x direction at the same rate with increasing De. In addition, no flow is observed across the $y = 0$ line in the streamlines of the $z = \pm 1$ planes. Figure 8-31 displays the change the streamlines between the $z = 0$ and $z = 1$ planes. Here the width of the region where the break in

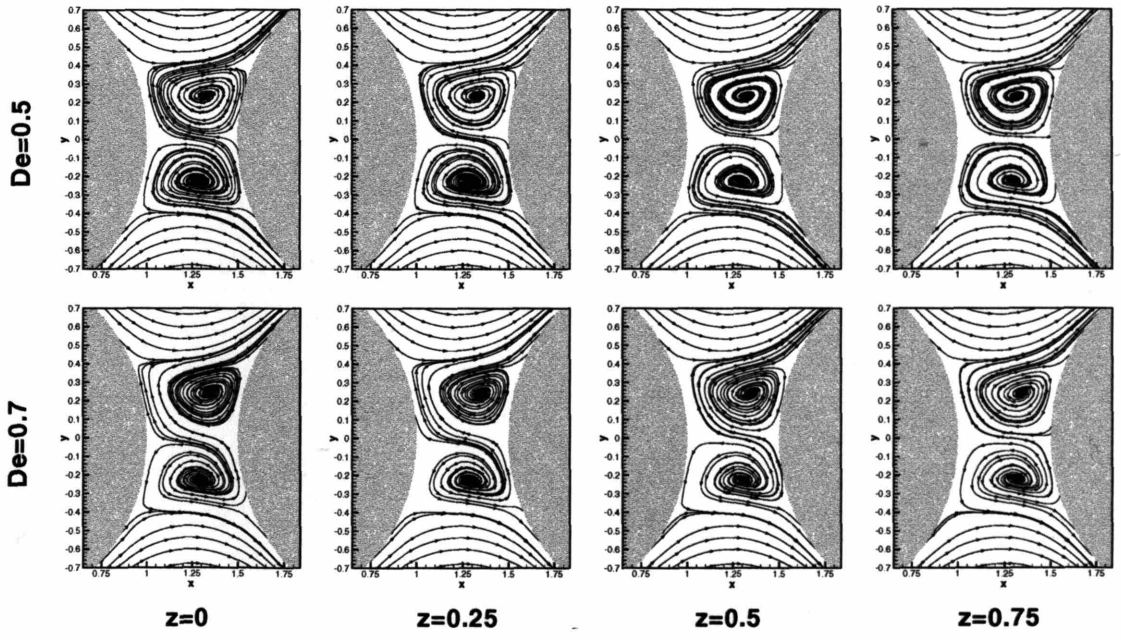


Figure 8-31: Streamlines in the $z = 0$, 0.25, 0.5, and 0.75 planes computed for the flow of an Oldroyd-B fluid with $\beta = 0.67$ and De of 0.5, and 0.7 in a periodic, linear array of cylinders with width of $W = 4R_c$.

the symmetry of the flow is present can be discerned. For the De $= 0.5$ case, the break in symmetry can be seen in the $z = 0.5$ plane, while for the De $= 0.7$ case, the break in symmetry can be seen at the $z = 0.75$ plane. Clearly, the width of the region where the break in symmetry occurs is growing with De.

While the xx, yx, and yy components of $\boldsymbol{\tau}_p$ exhibited virtually no differences compared to those of the geometry with periodic width, the results for the zz component of $\boldsymbol{\tau}_p$ is quite different. Recall that the zz component of $\boldsymbol{\tau}_p$ was found to be uniformly zero in the entire flow domain for all of the unbounded cases. Contours of τ_{pzz} in the $z = 0$ plane for the bounded wall case are shown in Fig. 8-32. Here the positive normal

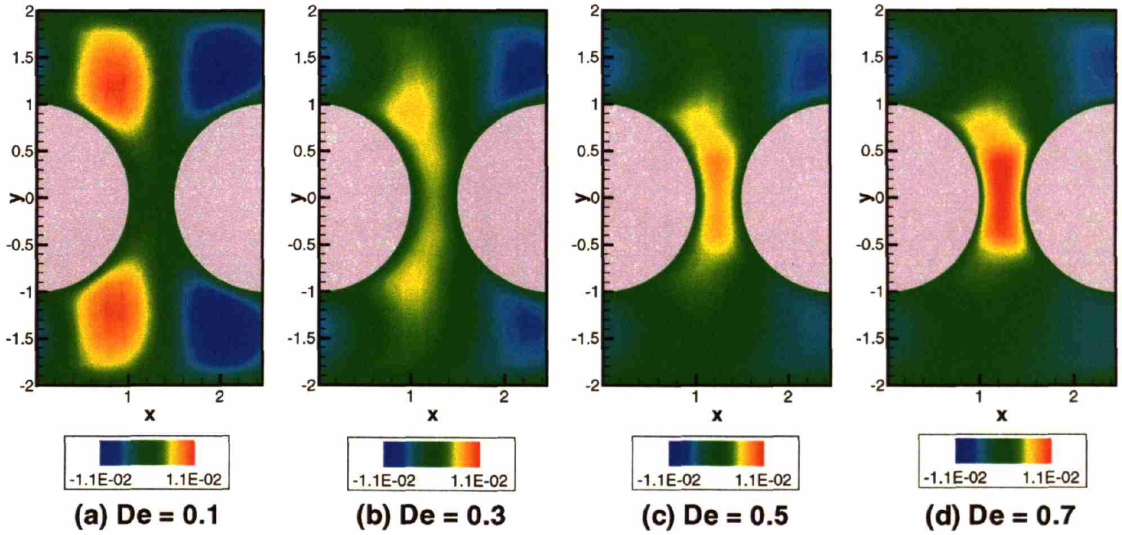


Figure 8-32: Contour plot of the zz component of $\boldsymbol{\tau}_p$ in the $z = 0$ plane for the flow of an Oldroyd-B fluid with $\beta = 0.67$ for a range of De in the periodic, linear array of cylinders with width of $W = 4R_c$. Flow is in the positive x direction.

stress in the z direction is shown to shift from just past the top edge of the cylinder to the gap between the cylinder as De increases. The positive region of τ_{pzz} also grows in intensity as it shifts into the gap. The regions of negative τ_{pzz} start just upstream of the top edge of the cylinder and move just above the top edge while decreasing in intensity as De is increased. The increase in normal stress in the z direction in the gap between the cylinders appears to be the cause of the movement and symmetry break of the vortices in the gap between the cylinders. Contours of $\boldsymbol{\tau}_{pzz}$ in the $z = -1.0$ and $z = -1.5$ planes are shown in Fig. 8-33.Moving outward from the $z = 0$ centerplane, the regions of positive $\boldsymbol{\tau}_{pzz}$ first shifts away from the $y = 0$ line in at $z = -1.0$. Closer to the

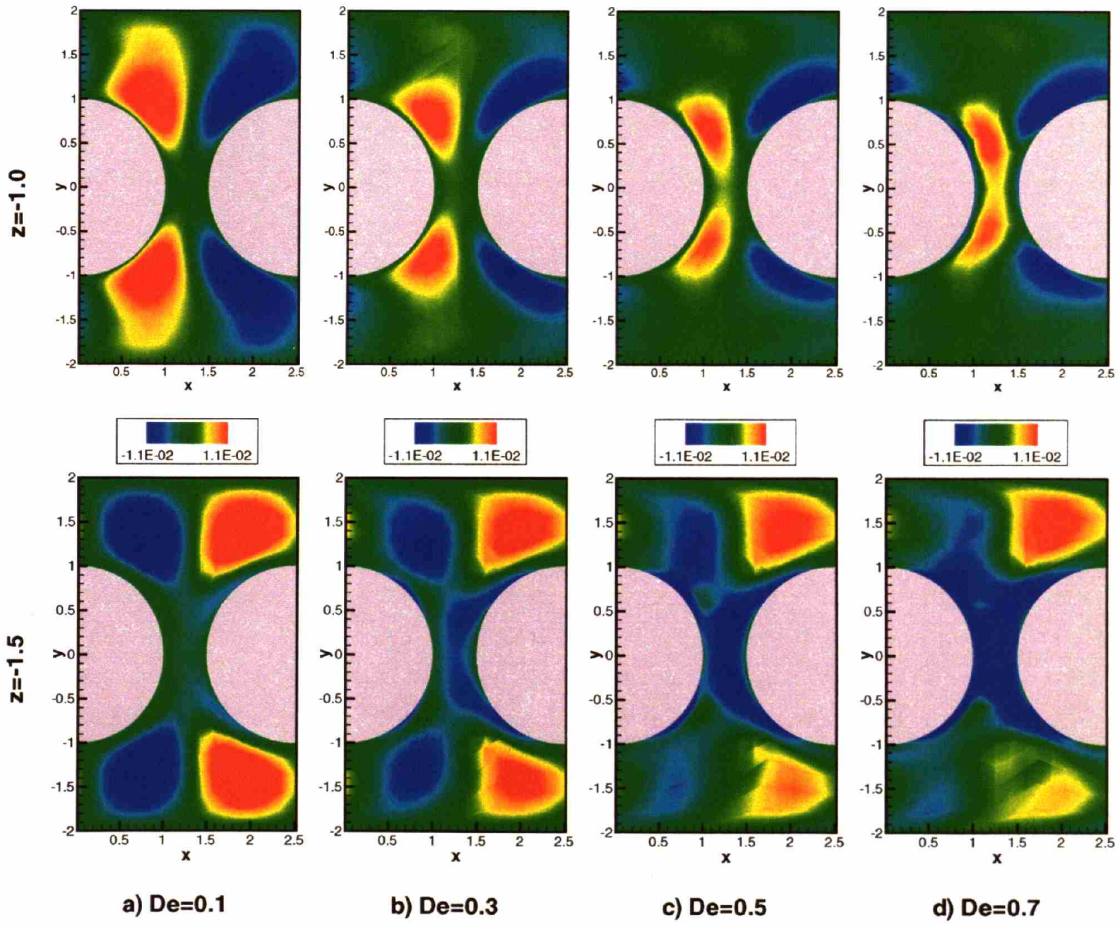


Figure 8-33: Contour plot of the zz component of $\boldsymbol{\tau}_p$ in the $z = -1.0$ and $z = -1.5$ planes for the flow of an Oldroyd-B fluid with $\beta = 0.67$ for a range of De in the periodic, linear array of cylinders with width of $W = 4R_c$. Flow is in the positive x direction.

bounding wall, at $z = -1.5$, $\boldsymbol{\tau}_{pzz}$ in the gap and surrounding region becomes negative. Similar growth in magnitude followed by a change of sign for $\boldsymbol{\tau}_{pzz}$ occurs for the negative regions near the upstream top and bottom edges of the cylinder. The positive regions of $\boldsymbol{\tau}_{pzz}$ remain relatively stationary in the $z = -1.5$ plane.

Effect of varying width of periodic domain

Since the instability in the periodic, linear array of cylinders with $L_c = 2.5R_c$ identified experimentally by Liu [55] and computationally by Smith et al. [73] has a period of

approximately $3R_c$, simulations of arrays with periodic widths of 2 and 3 were also performed to explore any similarities with the previously identified instability.

Simulation results for the $W = 2R_c$ case were very similar to those of the $W = 4R_c$ case. Steady state solutions were found for De up to 0.7, slightly less than in the $W = 4R_c$ case. Similar sensitivity to the timestep was also witnessed. Flow fields and stress fields also closely resembled those found in the $W = 4R_c$ case.

Simulation results for the $W = 3R_c$ case were somewhat unique compared to the other two cases. For this width, steady state solutions could only be found for De up to 0.5. However, flow fields and stress fields were found to be quite similar to the results of the other two cases. No unique flow structure appeared in any of the three different periodic widths simulated.

8.3 Conclusions

Simulations of the flow of a Newtonian fluid and an Oldroyd-B fluid in a periodic, linear array of cylinders were performed. The effects of modeling the geometry as infinite and unvarying in the z direction, as periodic in the z direction, and as periodic in the z direction with symmetry assumed about the $y = 0$ plane were studied. The effects of bounding the geometry in the z direction were also studied. For the flow of a Newtonian fluid, virtually no difference in the flow field in the $z = 0$ plane was found for the four cases. However, the bounding walls were shown to have a significant effect on the flow field less than R_c away from the $z = 0$ plane.

For the flow of an Oldroyd-B fluid, favorable comparisons were made to the work of Liu [55] and Smith et al. [73] for the geometry that was infinite and unvarying in the z direction. The effects of admitting solutions with variations in the z direction were then presented. The time-dependent simulations were shown to become increasingly sensitive to the timestep as De was increased, suggesting that an additional timescale dependent on De was introduced into the system with the modeling of the z direction. The flows

computed became increasingly asymmetric about the $y = 0$ plane, most evident in τ_{pyy} and v_z. The values of v_z suggest that the flow is moving back and forth along the axis of the cylinders forming one large recirculation cell with a higher magnitude below the cylinders than above. The values of v_z were however quite small when compared to v_x and v_y, and no evidence of such a cell was found in the streamlines of the flow. Simulations of the periodic width geometry with forced symmetry about the $y = 0$ plane were also performed for comparison with the full simulation. Sensitivity to the timestep was similar to that of the full simulation. Flow fields and stress fields compared well with the half of the flow domain in the full simulation that exhibited the lower magnitude v_z regions.

The effect of adding bounding walls to the geometry was quite dramatic with increasing De. While little motion was seen in the vortices in the gap between the cylinders for the periodic width case, including bounding walls caused the vortices in the $z = 0$ plane to grow significantly with increasing De and also broke the symmetry about the $y = 0$ plane. The normal stress in the z direction showed interesting behavior with the regions of positive τ_{pzz} on the downstream side of the outer edge of the cylinder shifting into the gap between the cylinders and increasing in intensity as De increased. The regions of negative τ_{pzz} found initially on the upstream side of the cylinder near the top and bottom walls shifted in between the cylinder and the wall and decreased in intensity as De increased. The motion and break in symmetry of the vortices was most likely caused by the movement and increase in intensity of the region of positive τ_{pzz}.

In order to attempt to compare flow structure found in the simulation to the instability identified by Liu et al. and Smith et al., simulations of the periodic linear array of cylinders with periodic width of 2, 3, and 4 were performed. The only significant effect found of varying the periodic width of the computational domain was a decrease in De at which the flow became unstable for the $W = 3R_c$ case relative to the $W = 2R_c$ and $W = 4R_c$ cases. However, no unique flow structure appeared in the simulation results for any of the three geometries with different periodic widths. Reasons for this are quite

likely that the simulations failed to reach steady state at Deborah numbers of almost half of that were the instability was identified. The refinement study carried out for the symmetric geometry suggest that increased resolution is required in the xy plane to obtain solutions at higher De. This is not surprising in that the two-dimesional simluations showed only qualitative agreement with the results of previous simluations that employed more highly refined meshes. Given hardware and operating system limitations, the meshes used for simulation of the full geometry were near the maximum possible size. Further studies with more resolved meshes should be completed once simulations can be performed on hardware with increased memory allocation limits.

Chapter 9

Conclusions and Further Work

9.1 Summary of work

The field of computational fluid mechanics of viscoelastic flows has been well explored in the three decades since its inception. Still, even with the vast amount of work detailed in the literature, a number of problems remain relatively untouched and a number of phenomena unexplained. This can be attributed to two areas in which research in the field is still quite prominent. The first area is that of the models used to describe the behavior of the polymer, be it based in the continuum approximation of the fluid, in the more direct description of the microstructure, or in the more exact modeling of the polymer molecules themselves. The second area is the numerical methods enabling the computational fluid mechanician to model ever more complicated flow geometries. Early simulation efforts in the field were concentrated around simple homogeneous flows and some coarse descriptions of heterogeneous flows. As the computational technology evolved and computer power improved, so did the desire to model more interesting geometries and relieve assumptions that were previously used to reduce problem size. In the past five years, a significant push towards the development of parallel solution techniques has appeared within the field, no doubt fueled by the appearance of relatively inexpensive clusters of PC's armed with free software allowing small groups of computational scientists to build

on-site parallel resources rather than wait for prescious supercomputer time as in the past. The creation of parallel solution technology has opened the door to the computation of complex geometries much more in tune with industrially relevant problems as well as vast increases in the level of detail included in the models of viscoelastic fluids.

The main goal of this thesis was to develop a robust, efficient simulation package to model three-dimensional viscoelastic flows. In order to accomplish this goal, improvements to the numerical methods and equation formulation were necessary to help reduce the overall size of the equation sets used to describe viscoelastic flows in three-dimensional geometries. In order to test their viability for use in reducing the overall size of the problem, concepts involving changing the formulation of the equations and the numerical methods used to find the solution to the equations were first implemented and analyzed in a previously developed two-dimensional finite element simulation package.

In Chapter 4 implementation and analysis was discussed of a formulation change involving the decoupling of the calculation of the velocity gradient interpolant equation and the momentum and mass continuity equations in the DEVSS-G formulation. Two different decoupled methods for computing the velocity gradient, one using a global least squares approximation and the other a local patch algorithm, were explored and the performance of each method was documented. While both methods in theory reduce to the true velocity gradient with mesh refinement, the patch algorithm was found to require significantly more mesh refinement than the global least squares approximation in order to attain equivalent accuracy in the solution. Comparison of the two methods taking into account the additional refinement requirements of the local patch algorithm made clear the superiority of the decoupled global least squares approximation for calculation of the velocity gradient interpolant.

Decoupling the calculation for the velocity gradient equation from the flow equations offered an even greater advantage than the initial splitting of the equation sets into two smaller problems. Decoupling the equations in this manner allows for optimal numerical methods to be applied to each equation set. The two-dimensional parallel

solver developed by Caola et al. ([16], [15]) demonstrated this advantage in the decoupling of the stress equations from the flow equations in the DEVSS-G formulation by allowing application of a specially tuned preconditioner to the flow equations. Here this idea is taken one step further by applying the Conjugate Gradient method to each component of the decoupled velocity gradient interpolant equation, which, in the decoupled form, are independent from one another and symmetric positive definite. Application of the CG method to each individual component reduces the size of the calculation for all of the components of the velocity gradient interpolant to less than one percent of the total calculation, making it in effect a free calculation. The total cost savings of the decoupled global least squares approximation is 1/2 the cost of the overall calculation using the coupled formulation.

In Chapters 5 and 6 the versatility and robustness of the decoupled form of the DEVSS-G equations were demonstrated through the addition and modification of the evolution equations describing the stress of the polymer as well as new physical quantities of the flow. Chapter 5 details the time-dependent, free-surface finite element method in which an evolution equation derived from the kinematic boundary condition is used to describe the height of the free surface as a function of time. This new evolution equation is incorporated into the decoupled formulation by simply adding an additional step to the time integration to evalulate the change in the height of the surface during the current timestep and then updating the element locations in the deformable region of the mesh. Application of the new equation in this manner requires no knowledge of the direct dependence of the system on changes in the new quantity, allowing for quick and easy implementation. The RK4 method is used to time integrate the stress equations in this implementation. The new evolution equation may not have similar stability to the stress equations; hence, analysis should be performed to determine an optimal time integrator for the new equation. In the case of the evolution equation for the height of the surface, stable time integration could be achieved with a forward Euler method interlaced with the RK4 method for integration of stress, reducing the overall cost of

adding the capability to handle unconfined geometries to the simulation package.

Incorporation of more advanced constitutive equations describing polymer stress relies much more heavily on the use of the decoupled equation set. For most continuum based constitutive equations, the dependence of the equations on the flow variables can be expressed explicitly, allowing for the coupled set of equations to be solved with Newton's method. However, in the more advanced constitutive equations, such as those derived from kinetic theory or those employing Brownian dynamics, the dependence of the stress on the flow cannot be explicitly written, greatly hindering the performance of Newton's method in locating the solution to the system. Chapter 6 details an example of incorporating relatively complicated closed-form constitutive equations within the decoupled equation formulation. The closed version of the Adaptive-Length-Scale (ALS-C) model developed by Ghosh et al. [34] is used to describe the Boger fluid used by Rothstein et al. [65] in the flow through a 4:1:4 axisymmetric contraction-expansion geometry. Although this equation is written in closed form, it includes a term that requires the evaluation of the eigenvector associated with the maximum eigenvalue of the rate-of-strain tensor of the local streamwise flow. Derivatives of this eigenvalue cannot be expressed analytically in terms of the velocity gradient. Use of this model in the decoupled equation formulation provided the first ever simulation of a dilute polymer solution model in a contraction-expansion geometry that exhibited the pressure drop enhancement with increasing viscoelasticity that has been well documented experimentally. A 4-mode FENE-P model was also used to represent the Rothstein fluid. Though its dependence on the flow field can be expressed analytically, the cost of computation with multimode models is typically prohibitive when using fully coupled equation sets, as the overall problem size grows considerably with the addition of each new mode. Incorporation of the 4-mode model within the decoupled equation formulation adds relatively little computational cost since the modes in the decoupled form that are time integrated with an explicit method are also decoupled from one another. Adding additional modes does still have an impact on the total amount of memory used in the calculation, but

this is a concern only for larger problems.

Once the formulation changes and numerical method improvements were implemented and well analyzed within the framework of the two-dimensional finite element package, development of the three-dimensional package was possible. Chapter 7 details the three-dimensional finite element simulation package for use with confined viscoelastic flows. The package utilizes the formulation and numerical methods developed in the two-dimensional package with modifications where necessary for the increase in dimensions of the physical problem. To make the package more robust, a number of different boundary conditions were included to model the different geometries used in polymer processing. To help reduce the burden associated with mesh refinement in three-dimensional meshes, a commercial meshing package utilizing o-grid refinement for localization of mesh refinement was employed. Furthermore, a parallel implementation of the three-dimensional simulation package was developed based on the two-dimensional parallel method of Caola et al. ([16], [15]).

As a test of the robustness of the three-dimensional method, simulations of the flow of Newtonian and Oldroyd-B fluids through a periodic, linear array of cylinders were performed and are detailed in Chapter 8. The study of this flow includes effects of modeling the cross section of the flow as (1) an infinite domain with no variation, (2) an infinite domain of periodic computational width, (3) an infinite domain of periodic computational width and a symmetric flow above and below the cylinders, and (4) a bounded domain with solid walls located 4 cylinder radii apart. Comparison with previous calculations for the Oldroyd-B flow in case (1) showed the same trend in the drag force on the cylinder with increasing viscoelasticity as well as in the size and shape of the vortices formed in the gap between the cylinders. The simulations were unable to capture the instability shown in [73], but the likely explanation is a need for even further increase in the mesh resolution, which was not possible in these simulations given the available hardware.

9.2 Future work

One of the long term research goals that this thesis contributes to is the development of a simulation package aimed at modeling industrial polymer processing technology including fiber spinning, film casting, and film blowing. While the three-dimensional simulation package developed here is a significant contribution toward this goal, there is still plenty of work needed to complete the goal. One of the capabilities that remains to be included in the three-dimensional simulation package is the modeling of free-surface boundaries and deformable regions of the computational domain. The two-dimensional, time-dependent, free-surface method developed in Chapter 5 is a good starting point for this endeavor. The free-surface boundary conditions employed in the two-dimensional method are directly extendable to three-dimensional geometries. Equations for updating the node locations of the elements in the deformable regions of the computational domain will require additional research, but the mapping equations employed in the two-dimensional method should prove to be a good starting point.

In addition to modeling unconfined geometries, additional physical models need to be added to the package to capture the behavior of the polymer. The most relevant physical effects that should be included in the simulation package are non-isothermal effects and crystallization effects (for crystallizing polymers such as polyethylene). Evolution equations describing the change in these physical quantities, such as the energy equation for modeling the local temperature of the polymer in the system, are easily implemented in the decoupled time-dependent formulation. This was demonstrated in the two-dimensional simulation package with the implementation of the evolution equation describing the motion of the free surface in Chapter 5 as well as with the implementation of the ALS-C and 4-mode FENE-P constitutive equations in Chapter 6.

In terms of this specific simulation package and the size of problems it is capable of solving, improvement of the memory scalability in the parallel machine is needed. More precisely, effort is needed to reduce the memory overhead associated with the overall problem that is carried on each machine in the parallel processor. The parallel method

implemented here has been shown to scale quite well with number of processors in the parallel machine, but as the overall problem size increases, so does the burden of overhead of book-keeping structures relating to the overall computational domain that are stored in memory on each machine. The hardware used in the larger calculations contained within this thesis consists of clusters of 32-bit PC's banded together with high-speed networking and configured to be run as a single parallel machine using software available on the internet. The limitation of this large parallel machine that was critical to the calculations presented here appeared in the memory allocation limits of 32-bit platforms. While the 32-bit platform is able to address a large total amount of memory commensurate with the total memory limitations of the motherboards (typically 64 gigabytes at present on server- and workstation-class machines), only around 2 gigabytes of memory can be allocated to a single process on the machine. While increasing the number of processors in the parallel machine reduces the size of the problem on each machine, the overhead on each machine associated with the overall problem persists. Hence, computation of very large problems is not possible due to the associated overhead. One possible course of action is to make software-based improvements that will reduce the overhead on each processor in the parallel machine. Another approach is to take advantage of improvements to the hardware that have been made available in the last year allowing allocation of virtually all available memory to a single process in the 32-bit platform. Intel's version of this architecture, the EMT64 class motherboards, is now included in the majority of workstation-class machines that are produced. Clusters composed of these machines with a sizable amount of physical memory in each machine will make possible the solution of much larger problems. This technology to address larger amounts of memory has been present in the 64-bit platforms for some time; however, the cost of these platforms are quite considerable when compared to their 32-bit counterparts. Since at least some overhead is likely always to be present on each of the machines in the large parallel calculations, it seems prudent to attack the problem of memory overhead from both the software and hardware fronts to allow for the most efficient simulation package

capable of simulating problems of adequate size to model industrial-type processes and handling ever increasingly complex constitutive equations.

Bibliography

[1] *ANSYS ICEM CFD: Meshing Tutorial Manual*, 4.3.1 edition, October 2003.

[2] Artist rendition of a 4dg fiber. Online image, http://www.clemson.edu/cucsm/concept/4dgfiber/Introd.htm, March 2005.

[3] M. A. Alves, F. T. Pinho, and P. J. Oliveira. The flow of viscoelastic fluids past a cylinder: finite-volume high-resolution methods. *Journal of Non-Newtonian Fluid Mechanics*, 97(2-3):207–232, 2001.

[4] Bala Ambravaneswaran, Scott D. Phillips, and Osman A. Basaran. Theoretical analysis of a dripping faucet. *Physical Review Letters*, 85(25):5332–5335, 2000.

[5] G. M. Amdahl. Validity of the single processor approach to achieving large scale computing capabilities. In *AFIPS conference proceedings*, pages 483–485, 1967.

[6] K. Arora, R. Sureshkumar, and B. Khomami. Experimental investigation of purely elastic instabilities in periodic flows. *J. Non-Newtonian Fluid Mech.*, 108(1-3):209–226, 2002.

[7] J. Azaiez, R. Guenette, and A. Ait-Kadi. Entry flow calculations using multi-mode models. *Journal of Non-Newtonian Fluid Mechanics*, 66(2-3):271–281, 1996.

[8] A. Baloch, P. Townsend, and M.F. Webster. On vortex development in viscoelastic expansion and contraction flow. *Journal of Non-Newtonian Fluid Mechanics*, 65(2-3):133–149, 1996.

[9] Brandon Max Baumert and Susan J. Muller. Axisymmetric and non-axisymmetric elastic and inertio-elastic instabilities in taylor-couette flow. *Journal of Non-Newtonian Fluid Mechanics*, 83(1-2):33–69, 1999.

[10] R. Byron Bird, Robert C. Armstrong, and Ole Hassager. *Dynamics of Polymer Fluids: Fluid Mechanics*, volume 1. John Wiley and Sons, New York, 1987.

[11] R. Byron Bird, Warren E. Stewart, and Edwin N. Lightfoot. *Transport Phenomena.* John Wiley and Sons, New York, 1960.

[12] D.V. Boger and K. Walters. *Rheological Phenomena in Focus.* Elsevier, New York, 1993.

[13] D.V. Boger and K. Walters. Experimental dilemmas in non-newtonian fluid mechanics and their theoretical resolution. *Korea-Australia Rheol. J.*, 12:27–38, 2000.

[14] R. A. Brown, M. J. Szady, P. J. Northey, and R. C. Armstrong. On the numerical stability of mixed finite element methods for viscoelastic flows governed by differential constitutive equations. *Theoretical and Computational Fluid Dynamics*, 5:77–106, 1993.

[15] A. E. Caola and R. A. Brown. Robust iterative methods for solution of transport problems with flow: a block two-level preconditioned schwarz-domain decomposition method for solution of nonlinear viscous flow problems. *Chemical Engineering Science*, 57(21):4583–4594, 2002.

[16] A. E. Caola, Y. L. Joo, R. C. Armstrong, and R. A. Brown. Highly parallel time integration of viscoelastic flows. *J. Non-Newtonian Fluid Mech*, 100:191–216, 2001.

[17] K. Chiba, R. Ishida, and K. Nakamura. Mechanism for entry flow instability through a forward-facing step channel. *J. Non-Newtonian Fluid Mech.*, 57(2-3):271–282, 1995.

[18] K. Chiba, T. Sakatani, and K. Nakamura. Anamalous flow patterns in viscoelastic entry flow through a planar contraction. *J. Non-Newtonian Fluid Mech.*, 36:193–203, 1990.

[19] K. Chiba, S. Tanaka, and K. Nakamura. The structure of anomalous entry flow patterns through a planar contraction. *J. Non-Newtonian Fluid Mech.*, 42(3):315–322, 1992.

[20] K. N. Christodoulou and L. E. Scriven. The fluid-mechanics of slide coating. *Journal of Fluid Mechanics*, 208:321–354, 1989.

[21] K. N. Christodoulou and L. E. Scriven. Discretization of free-surface flows and other moving boundary problems. *Journal of Computational Physics*, 99(1):39–55, 1992.

[22] P. J. Coates, R. C. Armstrong, and R. A. Brown. *Journal of Non-Newtonian Fluid Mechanics*, 42(1):141, 1992.

[23] B. Cockburn, G. E. Karniadakis, and C.-W. Shu. The development of discontinuous galerkin methods. In B. Cockburn, G. E. Karniadakis, and C.-W. Shu, editors, *Discontinuous Galerkin Methods: Theory, Computation and Application*, pages 3–50, New York, 2000. Springer.

[24] M.J. Crochet, A.R. Davies, and K. Walters. *Numerical simulation of non-Newtonian flow.* Rheology Series 1. Elsevier Science B.V., Amsterdam, The Netherlands, 1984.

[25] Marcela A. Cruchaga and Eugenio Onate. A generalized streamline finite element approach for the analysis of incompressible flow problems including moving surfaces. *Computer Methods in Applied Mechanics and Engineering*, 173(1-2):241–255, 1999.

[26] Benoit Debbaut, Thierry Avalosse, Joseph Dooley, and Kevin Hughes. On the development of secondary motions in straight channels induced by the second normal stress difference: experiments and simulations. *Journal of Non-Newtonian Fluid Mechanics*, 69:255–271, 1997.

[27] William M. Deen. *Analysis of Transport Phenomena.* Oxford University Press, New York, 1998.

[28] H.-S. Dou and N. Phan-Thien. Parallelized fvm algorithm for three-dimensional viscoelastic flows. *Computational Mechanics*, 30:265–280, 2003.

[29] Antonios K. Doufas, Anthony J. McHugh, and Chester Miller. Simulation of melt spinning including flow-induced crystallization part i. model development and predictions. *Journal of Non-Newtonian Fluid Mechanics*, 92(1):27–66, 2000.

[30] Antonios K. Doufas, Anthony J. McHugh, Chester Miller, and Aravind Immaneni. Simulation of melt spinning including flow-induced crystallization part ii. quantitative comparisons with industrial spinline data. *Journal of Non-Newtonian Fluid Mechanics*, 92(1):81–103, 2000.

[31] R. E. Evans and K. Walters. Flow characteristics associated with abrupt changes in geometry in the case of highly elastic flows. *Journal of Non-Newtonian Fluid Mechanics*, 20:11, 1986.

[32] R. E. Evans and K. Walters. Further remarks on the lip-vortex mechanism of vortex enhancement in planar-contraction flows. *Journal of Non-Newtonian Fluid Mechanics*, 32:95, 1989.

[33] Lars Herbert Genieser. *Stress- and velocity-field evolution in viscoelastic planar contraction flow.* Phd, Massachusetts Institute of Technology, 1997.

[34] I. Ghosh, Y. L. Joo, G. H. McKinley, R. A. Brown, and R. C. Armstrong. *Journal of Rheology*, 77(1):153, 2002.

[35] H. Giesekus. In E. H. Lee, editor, *Proceedings of the Fourth International Congress on Rheology*, volume 1, pages 249–266, New York, 1965. Wiley-Interscience.

[36] P.M. Gresho and R.L. Sani. *Incompressible Flow and the Finite Element Method Volume One: Advection-Diffusion.* John Wiley and Sons, Ltd, New York, 1998.

[37] P.M. Gresho and R.L. Sani. *Incompressible Flow and the Finite Element Method Volume Two: Isothermal Laminar Flow.* John Wiley and Sons, Ltd, New York, 1998.

[38] J. L. Gustafson. Reevaluating amdahl's law. *Communications of the ACM*, 31:532–533, 1988.

[39] M. T. Heath. *Scientific Computing: An Introductory Survey.* McGraw-Hill, Boston, 1997.

[40] Ole Hassager Henrik Koblitz Rasmussen. Three-dimensional simulations of viscoelastic instability in polymeric filaments. *Journal of Non-Newtonian Fluid Mechanics*, 82(2-3):189–202, 1999.

[41] E. Hinton and J.S. Campbell. Local and global smooting of discontinuous finite element functions using a least squares method. *International Journal for Numerical Methods in Engineering*, 8:461–480, 1974.

[42] Thomas J. R. Hughes. *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis.* Prentice-Hall, Inc., Englewood Cliffs, NJ, 1987.

[43] Bo-nan Jiang. *The Least Squares Finite Element Method.* Springer-Vertag, Berlin, 1998.

[44] Y. L. Joo, J. Sun, M. D. Smith, R. C. Armstrong, R. A. Brown, and R. A. Ross. Two-dimensional numerical analysis of non-isothermal melt spinning with and without phase transition. *Journal of Non-Newtonian Fluid Mechanics*, 102:37–70, 2002.

[45] D. D. Joseph. *Fluid Dynamics of Viscoelastic Liquids*, volume 84 of *Series in Applied Mathematics.* Springer-Verlag, Berlin, 1990.

[46] R. A. Keiler. *Journal of Non-Newtonian Fluid Mechanics*, 46(1):143, 1993.

[47] R. Keunings. Progress and challenges in computational rheology. *Rheol. Acta*, 29:556–570, 1990.

[48] R. Keunings. A servey of computational rheology. In D.M. Binding et al., editor, *Proceedings of the 13th International Congress on Rheology*, volume 1, pages 7–14, Glasgow, 2000. British Society of Rheology.

[49] Roger E. Khayat. Transient free-surface flow inside thin cavities of viscoelastic fluids. *Journal of Non-Newtonian Fluid Mechanics*, 91(1):15–29, 2000.

[50] K. Arun Kumar and Michael D. Graham. Buckling instabilities in models of viscoelastic free surface flows. *Journal of Non-Newtonian Fluid Mechanics*, 89(3):337–351, 2000.

[51] Youngdon Kwon. On instability of the doi-edwards model in simple flows. *Journal of Non-Newtonian Fluid Mechanics*, 88(1-2):89–98, 1999.

[52] J. Z. Liang, X. L. Sun, C. Y. Tang, and G. J. Tang. A study of the entry-region length of circular extrusion dies. *Journal of Materials Processing Technology*, 74(1-3):223–226, 1998.

[53] Yuanchang Liang, Alparslan Oztekin, and Sudhakar Neti. Dynamics of viscoelastic jets of polymeric liquid extrudate. *Journal of Non-Newtonian Fluid Mechanics*, 81(1-2):105–132, 1999.

[54] Alice W. Liu, David E. Bornside, Robert C. Armstrong, and Robert A. Brown. Viscoelastic flow of polymer solutions around a periodic, linear array of cylinders: comparisons of preditctions for microstructure and flow fields. *Journal of Non-Newtonian Fluid Mechanics*, 77:153–190, 1998.

[55] A.W. Liu. *Viscoelastic Flow of Polymer Solutions around Arrays of Cylinders: Comparison of Experiment and Theory.* PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 1997.

[56] X. L. Luo. Numerical simulation of weissenberg phenomena - the rod-climbing of viscoelastic fluids. *Computer Methods in Applied Mechanics and Engineering*, 180(3-4):393–412, 1999.

[57] Gareth H. McKinley, Robert C. Armstrong, and Robert A. Brown. The wake instability in viscoelastic flow past confined circular cylinders. *Philosophical Transactions of the Royal Society of London Series A-Mathematical Physical and Engineering Sciences*, 344(1671):265–304, 1993.

[58] E. Mitsoulis. Three-dimensional non-newtonian computations of extrudate swell with the finite element method. *Computer methods in applied mechanics and engineering*, 180:333–344, 1999.

[59] E. Mitsoulis and M. Beaulne. Numerical simulation of rheological effects in fiber spinning. *Advances in Polymer Technology*, 19(3):155–172, 2000.

[60] G. Mompean and M. Deville. Unsteady finite volume silumation of oldroyd-b fluid through a three-dimensional planar contraction. *Journal of Non-Newtonian Fluid Mechanics*, 72:253–279, 1997.

[61] J. Petera and V. Nassehi. Finite element modelling of free surface viscoelastic flows with particular application to rubber mixing. *International Journal for Numerical Methods in Fluids*, 23(11):1117–1132, 1996.

[62] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in Fortran: The Art of Scientific Computing.* Cambridge University Press, New York, 1992.

[63] Lidia M. Quinzani, Robert C. Armstrong, and Robert A. Brown. Birefringence and laser-doppler velocimetry (ldv) studies of viscoelastic flow through a planar contraction. *Journal of Non-Newtonian Fluid Mechanics*, 52(1):1–36, 1994.

[64] J.P. Rothstein and G.H. McKinley. Extensional flow of a polystyrene boger fluid through a 4:1:4 axisymmetric contraction/expansion. *Journal of Non-Newtonian Fluid Mechanics*, 86(1-2):61–88, 1999.

[65] J.P. Rothstein and G.H. McKinley. The axisymmetric contraction-expansion: the role of extensional rheology on vortex growth dynamics and the enhanced pressure drop. *Journal of Non-Newtonian Fluid Mechanics*, 98(1):33–63, 2001.

[66] Y. Saad. *Iterative Methods for Sparse Linear Systems.* SIAM, Philadelphia, 2nd edition, 2003.

[67] T.R. Salamon. *Finite Element Simulations of Steady Viscoelastic Free-Surface Flows.* Ph.d., Massachusetts Institute of Technology, 1995.

[68] Y. Shi. Performance quantification for parallel programs. In D. H. Norrie, editor, *Proceedings of the third international conference on finite elements in flow problems*, pages 204–215, Canada: Banff, AL, 1996.

[69] B. Smith, P. Bjorstad, and W. Gropp. *Domain decomposition: Parallel multilevel methods for elliptic partial differential equations.* Cambridge University Press, Cambridge, England, 1996.

[70] M. D. Smith, R. C. Armstrong, and R. A. Brown. Simulation of dynamics and structure formation in high-speed melt spinning. In *68th Annual Meeting of the Society of Rheology*, Galveston, Texas, 1997.

[71] M. D. Smith, R. C. Armstrong, R. A. Brown, and R. Sureshkumar. Finite element analysis of stability of two-dimensional viscoelastic flows to three-dimensional perturbations. *Journal of Non-Newtonian Fluid Mechanics*, 93(2-3):203–244, 2000.

[72] M.D. Smith. *Simulation of Nonisothermal and Time-dependent Viscoelastic Flows.* Ph.d., Massachusetts Institute of Technology, 2000.

[73] M.D. Smith, Y.L. Joo, R.C. Armstrong, and R.A. Brown. Linear stability analysis of flow of an oldroyd-b fluid through a linear array of cylinders. *J. Non-Newtonian Fluid Mech.*, 19:13–50, 2003.

[74] R. E. Smith. Algebraic grid generation. *Applied Mathematics and Computation*, 10:137–170, 1982.

[75] G. Strang. *Linear algebra and its applications.* Academic Press, Inc., New York, 2nd edition, 1980.

[76] J. K. Suen. *Multiscale Simulation of Viscoelastic Flows: Application to Kinetic Theory Mondels of Polymer Melts and Liquid Crystalline Polymers.* PhD thesis, Massachusetts Institute of Technology, 2002.

[77] J. Sun, M. D. Smith, R. C. Armstrong, and R. A. Brown. Finite element method for viscoelastic flows based on the discrete adaptive viscoelastic stress splitting and the discontinuous galerkin method: Davss-g/dg. *J. Non-Newtonian Fluid Mech*, 86:281–307, 1999.

[78] J. Sun, S. Subbiah, and J. M. Marchal. Numerical analysis of nonisothermal viscoelastic melt spinning with ongoing crystallization. *Journal of Non-Newtonian Fluid Mechanics*, 93(1):133–151, 2000.

[79] J. Sung, H. G. Choi, and J. Y. Yoo. Time-accurate computation of unsteady free surface flows using an ale-segregated equal-order fem. *Computer Methods in Applied Mechanics and Engineering*, 190(11-12):1425–1440, 2000.

[80] P. Szabo, J. M. Rallison, and E. J. Hinch. *Journal of Non-Newtonian Fluid Mechanics*, 72(1):73, 1997.

[81] M. J. Szady. *Finite element methods for time time dependent simulation of viscoelastic fluid flows.* PhD thesis, Massachusetts Institute of Technology, 1996.

[82] M. J. Szady, T. R. Salamon, A. W. Liu, D. E. Bornside, R. C. Armstrong, and R. A. Brown. A new mixed finite element method for viscoelastic flows governed by differential constitutive equations. *Journal of Non-Newtonian Fluid Mechanics*, 59(2-3):215–243, 1995.

[83] Shin-Ichiro Tanifuji, Tsuyoshi Kikuchi, Jun-Ichi Takimoto, and Kiyohito Koyama. Overall numerical simulation of extrusion blow molding process. *Polymer Engineering and Science*, 40(8):1878–1893, 2000.

[84] Milton Van Dyke. *An Album of Fluid Motion.* The Parabolic Press, Stanford, California, 1982.

[85] F. M. White. *Viscous Fluid Flow.* McGraw-Hill, 1974.

[86] S.-C. Xue, N. Phan-Thien, and R. I. Tanner. Numerical study of secondary flows of viscoelastic fluid in straight pipes by an implicit finite volume method. *Jounral of Non-Newtonian Fluid Mechanics*, 59:191–213, 1995.

[87] S. C. Xue, R. I. Tanner, and N. Phan-Thien. Three-dimensional numerical simulations of viscoelastic flows - predictability and accuracy. *Computer Methods in Applied Mechanics and Engineering*, 180(3-4):305–331, 1999.

[88] M. Yamaguchi, H. Miyata, V. Tan, and C. G. Gogos. Relation between molecular structure and flow instability for ethylene/alpha-olefin copolymers. *Polymers*, 43:5249–5255, 2002.

[89] T. Yamamoto, M. Ishiyama, M. Nakajima, K. Nakamura, and N. Mori. Three-dimensional viscoelastic flows through a rectangular channel with a cavity. *J. Non-Newtonian Fluid Mech.*, 114:13–31, 2003.

[90] O.C. Zienkiewicz and J.Z. Zhu. The superconvergent patch recovery and a posteriori error estimates. part 1: The recovery technique. *International Journal for Numerical Methods in Engineering*, 33:1331–1364, 1992.