Simulation techniques for viscoelastic fluids with zero solvent viscosity based on three field approaches

Dissertation

zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften

(Dr. rer. nat.)

Der Fakultät für Mathematik

der Technischen Universität Dortmund

vorgelegt von

Rida Ahmad

im Juli 2024

Dissertation

Simulation techniques for viscoelastic fluids with zero solvent viscosity based on three field approaches

Fakultät für Mathematik

Technische Universität Dortmund

Erstgutachter: Prof. Dr. Stefan Turek

Zweitgutachter: Privatdozent Dr. Andriy Sokolov

Tag der mündlichen Prüfung: 08 October 2024

Abstract: Solving viscoelastic fluid flow problems is challenging task due to their complex behavior, which involves the coupling of elastic and viscous stresses in a highly nonlinear manner. Additionally, numerically simulating pure polymer melts is particularly challenging due to the absence of solvent contributions to viscosity in the standard viscoelastic model. The absence of a diffusive operator in the momentum equation prevents the problem from being addressed in a decoupled manner and imposes limitations on the application of solution methods, generally rendering multigrid solvers impractical with a monolithic approach. This thesis aims to present a finite element method for solving the two-dimensional three-field Stokes flow for pure polymer melts, using the Elastic Viscous Stress Splitting (EVSS) and Tensor Stokes formulation. The formulation is expressed in terms of velocity, pressure, and the stress tensor. Both the EVSS and Tensor Stokes formulations help to reintroduce velocity coupling into the momentum equation by applying a change of variables in the standard viscoelastic formulation. This approach enables the problem to be handled in a decoupled manner and facilitates the application of multigrid solution methods using a monolithic approach. Nevertheless, this change of variables introduces additional terms with second-order velocity derivatives in the convective part of the constitutive equation for stress. To address this, the four-field approach is often employed, which includes the deformation tensor as an additional field to manage higher-order derivatives. The convective term is reformulated by taking into account the divergence-free nature of the velocity field, shifting the higher-order derivatives to the test function in the weak formulation thus maintaining the problem size to three field. The velocity, pressure, and stress are discretized using the higher-order stable FEM triplet (Q_2, P_1^{disc}, Q_3). The proposed scheme is evaluated using the Oldroyd-B, Giesekus, and PTT exponential fluids, employing both decoupled and monolithic solution approaches. Numerical results are obtained for a four-to-one curved contraction, for highly viscoelastic fluids with the aim to achieve results at relatively large values of the relaxation parameter λ and observe the shear-thinning effect w.r.t. the relaxation parameter λ.

Key words: Elastic Viscous Stress Splitting, Tensor Stokes formulation, Viscoelastic fluids, Pure polymer melts, Finite Element Method, Decoupled and monolithic approaches, Shear thinning effect, Multigrid.

Rida Ahmad
University of Dortmund, Germany

To my parents,
husband and siblings

Acknowledgement. First and foremost, I want to express my deepest regard to my supervisor, Prof. Dr. Stefan Turek. Working under his valuable expertise in his esteemed research chair has been a dream come true. His remarkable accomplishments in the field of non-Newtonian fluid flow problems as well as internationally acknowledged achievements in academia have been truly inspiring. Despite his busy schedule as Dean, he was always available whenever needed, offering insightful feedback that pushed me to enhance my thinking and elevate my research. His optimism, effective communication, valuable guidance and constructive feedback made a significant difference that pushed the research work to the current state. These important factors have greatly contributed to the completion of my thesis and made my PhD journey a remarkable experience. Prof. Turek, thank you once more for your guidance, confidence in me, and for accepting to review my dissertation.

I am immensely grateful to Mr. Peter Zajac for generously sharing his valuable expertise in FEATFLOW3. The clear explanations and patient support have significantly contributed to my growth in learning and implementing my problem in FEATFLOW3. I am deeply grateful to Dr. Patrick Westervoss for the insightful discussions and guidance provided regarding viscoelastic flows. His expertise and willingness to share knowledge have been instrumental in shaping my understanding of this complex subject. Thank you, Mr. Zajac and Dr. Westervoss, for your significant contributions to my research journey in viscoelastic flows.

I would like to extend my thanks to all LSIII colleagues and secretaries for fostering a positive and collaborative working environment. I would also like to gratefully acknowledge the support by LSIII and LiDO3 team at ITMC, TU Dortmund University, Germany. I am extremely grateful to the DAAD "Deutscher Akademischer Austauschdienst" for the financial support.

Finally, I want to thank my parents and my dear husband Enan Khan for continuously supporting me and believing in me.

Dortmund, 2024
Rida Ahmad

Table of Contents

1

Introduction

1.1 General overview of Newtonian fluids

We encounter flow processes in almost all areas of our life. For instance, the air around us is fluid, our body is composed of 80% of fluids and the Earth's surface consists of 71% water. Fluid is everywhere and it is an essential part of our everyday life e.g. blood flow in human body, air pollution, ventilation systems in buildings, rain, floods, propulsion systems and so on. Water and air are the simplest fluids called Newtonian fluids. In Newtonian fluids, the stress is directly and linearly proportional to the strain rate. In non-Newtonian fluids, either there exists a non-linear relation between the shear stress and shear rate or the viscosity depends upon time or the deformation history. Examples of non-Newtonian fluids include toothpaste, ketchup, rubber, honey etc.

The presence of fluids every where around us and their importance in our daily life makes their study imperative for the development and progress of human life. The interesting fluid behavior can be studied either with the help of experiments or numerical simulations. In this thesis, we are interested to investigate the behavior of various fluids with the help of mathematical modeling and their numerical simulations. The fluid motion is governed and represented by the mathematical models (partial differential equations) that constitute the conservation of mass and momentum. The simplest mathematical model that governs the fluid flow is represented by the Navier-Stokes equations that governs the conservation of mass and momentum is given as,

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho \left(\mathbf{u} \cdot \nabla \right) \mathbf{u} + \nabla p = \nabla \cdot \boldsymbol{S} \tag{1.1}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{1.2}$$

In equations (1.1) and (1.2), the fluid density, velocity, pressure and extra stress tensor are represented by ρ, $\mathbf{u}$, p and $\boldsymbol{S}$, respectively.

1.2 Viscoelastic fluids

In particular, we are interested to study the viscoelastic class of non-Newtonian fluids. These fluids couple the elastic stress with the Newtonian viscous stress, and are highly non-linear in nature. Depending upon the stress, such fluids either behave as solid or liquid. Examples of viscoelastic fluids include polymers and rubber melts. Polymer melts, having the viscoelastic properties produce noteworthy effects during their flow that distinguish them from the Newtonian fluids. The interesting flow behaviors include, for instance, the rod-climbing effect and die swell. The rod-climbing effect also called the Weissenberg effect shown in Fig. 1.1 was first reported by Weissenberg [33]. It can be observed that the Newtonian fluid when stirred or rotated does not climb the rod whereas the viscoelastic fluid climbs up the rotating shaft. The real life example of this behavior is the whipping cream that tends to climb up the rod of a beater. Die swell, also called as Barus effect

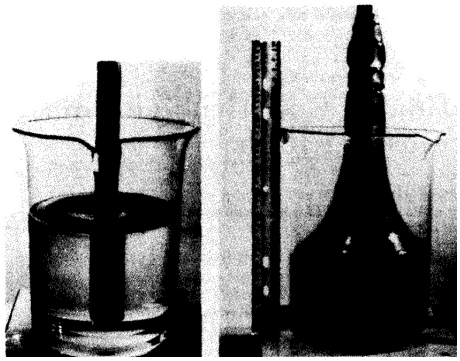


Fig. 1.1: **Rod climbing effect.** The Newtonian fluid does not climb (*Left*). Viscoelastic polyarcylamide solution climbs up the rotating shaft (*Right*). [11]

or extrude effect is another extremely interesting behavior exhibited by the viscoelastic fluids. When the viscoelastic fluid comes out of a tube, the diameter of the viscoelastic jet is usually greater than the diameter of the tube. This increase can be upto three times the diameter of the channel. However, the Newtonian jet exhibits a very slight swelling. This phenomena can be observed in Fig. 1.2. The interesting effects produced by the viscoelastic fluids make their study imperative for understanding the phenomenons occurring in daily life or at industrial level. However, the numerical solutions of the viscoelastic fluids are quite expensive and exhibit several challenges. The solution of the viscoelastic flow problems require the coupling of the incompressible Navier Stokes equation with the constitutive equation that exhibits the viscoelastic behavior of the fluid by providing a relationship between the stress tensor and the velocity gradient. The challenge is to handle

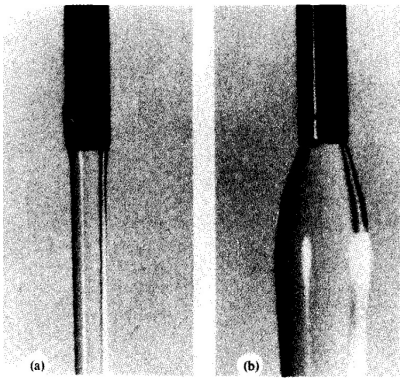


Fig. 1.2: **Die swell phenomena.** The Newtonian fluid does not swell (*Left*). The viscoelastic fluid emerging from the tube swells significantly (*Right*) [11].

the complex material behavior of the viscoelastic fluids that is either represented by the differential or the integral equation.

Therefore, developing reliable and efficient numerical techniques for simulating such fluid flows is a challenging task. By employing a variety of numerical techniques, researchers have been actively investing in and reporting the development of a stable, accurate, and efficient numerical algorithm. These numerical methods include, finite element method, finite volume method and spectral methods. To our interest and keeping in view the importance and contributions to the scientific world we have selected finite element method for our numerical procedures. The finite element method is an effective tool that has found widespread application, including, but not limited to, electromagnetism, fluid mechanics, structural mechanics, biology, chemistry, financial modelling, and superconductivity. Everything from stress analysis of a human tooth to the design of an aeroplane wing has been studied using the finite element approach. In 1940's and 1950's, structural engineers were the first to employ the methodology; Mathematicians only started to show an interest in its analysis and application in the late 1960's [39]. In this work, we use finite element method to numerically approximate the solution of the incompressible, isothermal, stationary and non-stationary viscoelastic formulations with the particular choice of the constitutive equations.
In the recent years, researchers have actively reported the accurate and robust numerical techniques for the finite element method solution of the viscoelastic flows. Following are the few challenges that must be addressed while obtaining the solution of the three field viscoelastic Stokes flow;

1. The finite element spaces for the velocity and pressure need to be selected with care, in order to satisfy the compatibility condition between the velocity and pressure space called inf-sup condition or LBB (Ladyzhenskaya-Babuska-Brezzi) condition [14].

2. The presence of the convective term $\mathbf{u} \cdot \nabla \mathbf{S}$ in the constitutive equation requires numerical schemes that are suitable for transport dominated problems.
3. For pure polymer melts, where the solvent contribution in the momentum equation is absent, an additional compatibility condition between the finite element spaces for the velocity and stress must be met.

While working with the finite element method the choice of the stable finite element pair is a key factor in obtaining the accurate results. In the three field Stokes problem, the FEM spaces used to discretize the velocity, pressure and stress must be compatible according to the well-known inf-sup condition or LBB condition [14]. There are a vast number of finite element spaces satisfying the inf-sup condition for the velocity and pressure coupling. However, for the extra stress tensor the choices are limited. Fortin and Pierre [24] have suggested that if the continuous FEM is used to approximate the stress tensor, then the stress space should be larger than the velocity space i.e., the number of stress local degrees of freedom must be larger than the local number of degrees of freedom for the velocity space. Furthermore, their study suggests that if the stress tensor is approximated by the discontinuous FEM then the deformation tensor must be the member of the same space as stress tensor. The first idea suggested by Fortin and Pierre supports the work of Marchal and Crochet [42]. They worked with the stable finite element pair by using the continuous FEM. They divided the stresses into the several bilinear sub-elements i.e., they enriched the local degrees of freedom for the stress space by applying the subcell discretization. Marchal and Crochet obtained excellent results with such subcell discretization. Moreover, the successful results obtained by Fortin and Fortin [22] endorse the idea of discontinuous FEM proposed by Fortin and Pierre [24]. Therefore, the finite element spaces for the velocity, pressure and stress cannot be chosen arbitrarily rather they should satisfy the inf-sup condition. Keeping in view the importance of the stable finite element pair, the researchers have actively worked in the field and during the last years various successful stable finite element pairs have been developed [23, 25, 55].

The next challenge faced by the scientific community while working with the three field viscoelastic formulation is the appearance of the convective term in the constitutive equation. As the relaxation parameter λ increases, the convective term of the constitutive equation becomes more dominant and the standard Galerkin formulation starts to have problems. Therefore, this term requires special treatment. The convective contribution in the constitutive equation can be stabilized via well-known streamline upwind/Petrov–Galerkin (SUPG) method, Streamline Upwind (SU), discontinuous Galerkin and the edge oriented (EO) stabilization. The streamline-upwind/Petrov-Galerkin (SUPG) method introduced by [2] was first employed to the viscoelastic fluid by Marchal and Crochet [42]. It was

observed that the SUPG method produce oscillations in the stress near the singularities and therefore a modified version of the SUPG was proposed by Marchal and Crochet [42] called the streamline-upwind SU method. The modification led to the application of the upwind term only to the convective part of the constitutive equation. The SU method helped Marchal and Crochet in obtaining the smooth solutions without stress singularities. However, this method is inconsistent because the upwinding term targets only the convective part of the constitutive equation. An alternative to SUPG and SU stabilzation is discontinuous Galerkin (DG) method. Fortin and Fortin [22] were the first to introduce the DG method in context of viscoelastic fluids. The present study is in the favor of the Edge-Oriented stabilization. The main idea of the Edge Oriented stabilization is to augment the original finite element discretization by an interior penalty term involving the jump of the solution gradient over element edges E (with h_E denoting the length of the edge)[61]. The idea is to increase the robustness of the Galerkin approximation.

The interest is to simulate the viscoelastic fluids with pure polymer viscosity contribution i.e., zero solvent viscosity. The simulation of such fluids is a challenging task because it leads to the absence of the velocity coupling in the momentum equation and therefore it gets very difficult to handle such models numerically. While working with the standard three field Stokes problem, stabilization techniques become inevitable. Moreover, the vast demand of simulating such fluids has motivated scientists to invest more into the simulation techniques of viscoelastic fluids with zero solvent viscosity. The exploration of the hidden rheology within the extra-stress tensor can be employed to develop the mathematical models such that the velocity coupling in the momentum equation can be recovered for the case of zero solvent viscosity that means the momentum equation contains the diffusive term. Mendelson et al. [43] first proposed the idea of splitting extra stress tensor into its elastic and viscous components. This splitting is called Elastic Viscous Stress Splitting (EVSS). Beris et al. [9] used the EVSS formulation to obtain the results for the viscoelastic fluids using finite element method. Rajagopalan et al.[53] employed the concept of the EVSS formulation and presented a four field algorithm to obtain the numerical solutions of the viscoelastic fluids. The success of EVSS algorithm presented by [53] led to the development of the modified form of the EVSS that is called discrete EVSS or DEVSS suggested by Guenette and Fortin [28]. The essence of the EVSS formulation or its discrete version lies in the splitting of the extra stress into its elastic and viscous contributions that results in the addition of the diffusive operator in the momentum equation. Such splitting helps to obtain the solution of the viscoelastic fluids for the case of zero solvent viscosity. Recently, Westervoss [66] has introduced a novel splitting technique of the extra-stress tensor called the Tensor Stokes formulation. The extra-stress tensor has been split into the product of deformation tensor and nonsymmetric tensor based viscosity with the main advantage to explicitly describe the complex material properties

and behavior by means of a viscosity tensor. The Tensor diffusion technique provides the opportunity to reduce the three field viscoelastic formulation to the two field formulation only in $(\mathbf{u}, p)$, with the modified Stokes part. This can be done by introducing the diffusion viscosity (that explicitly contains the properties of the viscoelastic model) into the momentum equation. The constitutive equation can now be ignored. The success of the Tensor Stokes formulation provides the opportunity to employ such extra-stress decomposition in the viscoelastic formulations in various directions. Our focus in this thesis is to analyze the three field viscoelastic formulation with zero solvent viscosity. Similar to the well-known EVSS formulation, the novel splitting of the extra-stress tensor called the Tensor diffusion approach has the potential to recover the velocity coupling into the momentum equation for the zero solvent viscosity in the three field formulation. In this thesis, we have employed the EVSS and Tensor Stokes formulation for our numerical simulations.

The numerical treatment of the viscoelastic fluids with zero solvent viscosity using the well-known EVSS formulation has been reported by various scientists. On one hand where the velocity coupling in the momentum equation is recovered, some new terms appear in the constitutive equation that involves the convected derivative of the rate of deformation tensor, which means that the second order velocity derivatives needs to be treated in the constitutive equation. The appearance of the second order velocity derivatives is difficult to handle numerically and therefore require special treatment. In order to circumvent this problem Rajagopalan et al.[53] introduced an additional unknown the rate of deformation tensor to treat the derivative of the strain tensor seperately. A modified form of the EVSS was also introduced by Guenette and Fortin [28] which appears to be the discrete version of the EVSS formulation proposed by Rajagopalan et al. [53]. The introduction of an additional variable in the EVSS formulation leads to the four field formulation from the three field formulation, and thus the problem size is increased. This is numerically cumbersome and expensive because now one has to treat four flow variables instead of three. The Tensor Stokes formulation could be employed in the same pattern of introducing an additional variable like in the EVSS formulation, and therefore poses the same problem. In this thesis, we have employed a novel technique and reformulated the convective term to deal with the second order velocity derivatives appearing in the constitutive equation of the EVSS and Tensor Stokes formulation. The technique helps to retain the problem size to three field formulation. The overall advantage of the present study is to deal with the viscoelastic fluids with zero solvent viscosity using the well-known EVSS formulation and novel Tensor-Stokes formulation without increasing the problem size. The aim is to explore the possibilities and benefits of using a three-field formulation with different constitutive equations, having as unknowns the extra-stress tensor or the elastic part of the stress tensor or the tensor based viscosity coupled with the Stokes equations.

1.3 Structure of the thesis

Having explained the motivation of the research work, the layout of the thesis is outlined below:

Starting with the Stokes flow, *Chapter 2* is dedicated to describe the standard three field differential viscoelastic formulation and its limitations for the pure polymer melts. To address these limitations, the chapter introduces stress splitting as a method to overcome shortcomings in standard viscoelastic formulations. The Elastic Viscous Stress Splitting (EVSS) and Tensor Stokes three field differential viscoelastic models for various fluids of interest are mathematically expressed and described. Integral viscoelastic models are also considered as part of the discussion in this chapter. The stress splitting approach, which was originally developed for differential viscoelastic models, is employed to the Integral viscoelastic models.

Chapter 3 deals with the numerical challenges encountered in obtaining the solution for the EVSS and Tensor Stokes differential models. Having introduced the stress splitting in the standard differential viscoelastic formulation, the velocity coupling into the momentum equation is recovered. However, the second order derivatives of the velocity appear in the constitutive equation in both the EVSS and Tensor Stokes formulation. The chapter deals with novel remedy to encounter the second order velocity derivatives.

The system of equations is solved using finite element method described in *Chapter 4*. The weak formulation and finite element discretization for the three field viscoelastic formulation is presented. Implicit Euler method is employed for the time discretization. The Edge oriented stabilization technique is also the part of this chapter.

In *Chapter 5*, the description of the numerical solvers is presented. Fixed point method is used as an outer non-linear solver, where UMFPACK or iterative methods (basic iterative methods, Krylov subspace methods or Multigrid methods) can be used as linear solver inside outer non-linear sweep. The chapter also includes the solution methods for the viscoelastic three field Stokes formulation via monolithic and decoupled/operator-splitting approaches. The solution approaches for the EVSS and Tensor Stokes viscoelastic formulation for various fluids of interest are mathematically expressed in detail.

Chapter 6 includes the numerical results starting with the validation of the code by employing the three-field stationary Newtonian formulation for the "flow around the cylinder" benchmark and the "stick-slip problem". The numerical simulations for the simplest Oldroyd-B fluid are performed on the square domain for the standard viscoelastic formulation. The corresponding errors between the analytic and numerical results are computed. The stationary EVSS and Tensor Stokes formulation are tested for the Oldroyd-B model on the square domain utilizing the uniform and perturbed variant of the mesh and

employing both the monolithic as well as decoupled solution technique. The error computation between the analytic and numerical results are presented for increasing values of the relaxation parameter λ.

Chapter 7 deals with the detailed numerical study performed on four-to-one contraction configuration for the complex viscoelastic fluids. The aim is to simulate the higher value of the relaxation parameter λ for the complex geometry and fluids of interest. The chapter includes the steady state numerical simulations using the monolithic approach for the Oldroyd-B, Giesekus and PTT-exponential model, for both the EVSS and the Tensor Stokes viscoelastic formulations. A comparison of computational cost between the monolithic and decoupled EVSS formulation is also carried out in this chapter. The achievement of our objectives in non-stationary simulations is facilitated by employing the adaptive time stepping technique using the EVSS formulation with PTT-exponential model. The adaptive time stepping technique helped us to obtain solution at higher values of the relaxation parameter λ efficiently. A brief study of the Multigrid solver for the Stokes part is also conducted and provided in this chapter.

In *Chapter 8* the comprehensive summary of the research findings and closing remarks are added.

2

Governing equations

2.1 Newtonian Stokes flow

Let us consider the general form of incompressible Navier-Stokes equation that arises by the coupling of momentum and continuity equation. The momentum equation (2.1a) is balanced by inertial forces on the left hand side and viscous forces on the right hand side. Equation (2.1b) shows the conservation of mass. The detailed derivation of the problem can be found in [38]. Thus, the governing equations are:

$$\rho\frac{\partial \mathbf{u}}{\partial t} + \rho\left(\mathbf{u}\cdot\nabla\right)\mathbf{u} = \eta\nabla^2\mathbf{u} - \nabla p + \mathbf{F} \tag{2.1a}$$

$$\nabla\cdot\mathbf{u} = 0 \tag{2.1b}$$

In the set of equations (2.1), $\mathbf{u}$ is the fluid velocity, p is the fluid pressure field, $\mathbf{F}$ refers to the external forces, ρ and η are the constants that correspond to the density and viscosity of the fluid. In the above mentioned formulation, the specific case of the creeping flows is obtained by neglecting the inertial forces. Thus, the steady state incompressible Stokes flow reads as, follows:

$$\eta\nabla^2\mathbf{u} - \nabla p = 0 \tag{2.2a}$$

$$\nabla\cdot\mathbf{u} = 0 \tag{2.2b}$$

Problem (2.2) can be expressed in the form of the constitutive law for the Newtonian fluid in terms of extra-stress tensor as follows:

$$\nabla\cdot\boldsymbol{S} - \nabla p = 0 \tag{2.3a}$$

$$\nabla\cdot\mathbf{u} = 0 \tag{2.3b}$$

where $\boldsymbol{S} = 2\eta\mathbf{D}(\mathbf{u})$ is the constitutive law for the Newtonian fluid. $\mathbf{D}$ is the deformation tensor given as $\mathbf{D}(\mathbf{u}) = {}^1\!/_2\left(\nabla\mathbf{u} + \nabla\mathbf{u}^T\right)$. Problem (2.3) represents the balance of forces in

creeping flows. In order to explore the viscoelastic fluids the momentum and continuity equation are coupled with the nonlinear constitutive equation involving velocity and stress that depicts the elastic behavior of the fluid. The viscoelastic fluid models are discussed in the following sections.

2.2 Differential viscoelastic formulations

In this section, we are aiming to outline the differential viscoelastic formulations whose key feature is the relaxation parameter λ. It is the time scale that depicts the time needed by the viscoelastic fluid to recoil back to its original state after deformation, irrespective of the initial state. In the viscoelastic model, if the relaxation parameter λ is set to zero, the Newtonian flow is recovered. Thus, the relaxation parameter λ holds a very significant position, as it depicts the degree of elasticity in the viscoelastic material. In the viscoelastic model, as the degree of the elasticity is increased, so does the numerical complexity, which frequently leads to computational failures, even at lower relaxation parameter. This challenge arises from the issue of adequately approximating the exponentially increasing stress profiles using polynomial functions, which are typically employed in finite element discretizations. In the similar manner, large relaxation parameter increases the challenges of the discrete problem, as it enhance the influence of the convective term in the upper convected time derivative of the differential constitutive equation. The detailed study of the underlying material behavior can be found in [16, 4, 37, 41, 36]. The viscoelastic formulations consists of the momentum, continuity and constitutive equation, where the constitutive equation can be expressed in both differential and integral form.
In the following subsections, the differential form of the standard viscoelastic formulation, Elastic viscous stress splitting (EVSS) and Tensor Stokes formulation are discussed.

2.2.1 Extra stress tensor formulation

In this work, we shall consider the incompressible, stationary/non-stationary, three field viscoelastic Stokes formulation. The equation of motion for the Stokes flow is characterized by extremely low Reynolds number such that the viscous forces outweigh the inertial forces. Mathematically, the formulation is expressed as follows:

$$\rho\frac{\partial \mathbf{u}}{\partial t} - 2\eta_s \nabla \cdot \mathbf{D}(\mathbf{u}) - \nabla \cdot \boldsymbol{S} + \nabla p = 0 \tag{2.4a}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{2.4b}$$

$$\lambda \overset{\nabla}{\boldsymbol{S}} + f(\boldsymbol{S}, \lambda)\,\boldsymbol{S} + h(\boldsymbol{S}) = 2\eta_p \mathbf{D}(\mathbf{u}) \tag{2.4c}$$

Equations (2.4a) and (2.4b) are Stokes equations coupled with an additional constitutive stress equation (2.4c). In the above mentioned formulation, $\mathbf{u}, p$ and $\boldsymbol{S}$ are the flow field variables that indicate the fluid's velocity, pressure and extra stress tensor, respectively. Tensor $\mathbf{D}(\mathbf{u})$ is the deformation tensor, which is the symmetric part of the velocity gradient. The assumption of the incompressible flows leads to a constant density ρ, which is taken as 1. The constants η_s and η_p are the solvent and polymer viscosity, respectively. Having zero contribution from the solvent viscosity i.e. η_s=0 results in the pure polymer melts, λ is the relaxation parameter that characterizes the elastic properties of the fluid. The term $\overset{\nabla}{\boldsymbol{S}}$ is the upper-convected time derivative defined by following:

$$\overset{\nabla}{\boldsymbol{S}} = \frac{\partial \boldsymbol{S}}{\partial t} + (\mathbf{u} \cdot \nabla)\, \boldsymbol{S} - \nabla \mathbf{u}^T \cdot \boldsymbol{S} - \boldsymbol{S} \cdot \nabla \mathbf{u} \tag{2.5}$$

The two functions $f(\boldsymbol{S}, \lambda)$ and $h(\boldsymbol{S})$ in equation (2.4c) defines the particular viscoelastic fluid model. The constitutive equation for the simplest viscoelastic Oldroyd-B and UCM (Upper convected Maxwell) fluid [44] is obtained by setting $f(\boldsymbol{S}, \lambda)$=1 and $h(\boldsymbol{S})$=0. Thus, equation (2.4c) becomes:

$$\lambda \overset{\nabla}{\boldsymbol{S}} + \boldsymbol{S} = 2\eta_p \mathbf{D}(\mathbf{u}) \tag{2.6}$$

The following Giesekus model [26] is achieved by setting $f(\boldsymbol{S}, \lambda)$=1 and $h(\boldsymbol{S})$=$\frac{\alpha\lambda}{\eta_p}(\boldsymbol{S} \cdot \boldsymbol{S})$;

$$\lambda \overset{\nabla}{\boldsymbol{S}} + \boldsymbol{S} + \frac{\alpha\lambda}{\eta_p}(\boldsymbol{S} \cdot \boldsymbol{S}) = 2\eta_p \mathbf{D}(\mathbf{u}) \tag{2.7}$$

where $\alpha \in [0, 1]$ refers to the mobility factor. In equation (2.4c), setting $f(\boldsymbol{S}, \lambda)$=*function* and $h(\boldsymbol{S})$=0, one get the Phan-Thien-Tanner (PTT) model in it's linear and exponential form.

$$\lambda \overset{\nabla}{\boldsymbol{S}} + g(\mathrm{tr}\boldsymbol{S})\boldsymbol{S} = 2\eta_p \mathbf{D}(\mathbf{u}) \tag{2.8}$$

The stress function $g(\mathrm{tr}\boldsymbol{S})$ is either linear [51],

$$g(\mathrm{tr}(\boldsymbol{S})) = 1 + \frac{\lambda\kappa}{\eta_p}(\mathrm{tr}(\boldsymbol{S})) \tag{2.9}$$

or it takes the exponential form as proposed in [50]

$$g(\mathrm{tr}(\boldsymbol{S})) = \exp\left[\frac{\lambda\kappa}{\eta_p}(\mathrm{tr}(\boldsymbol{S}))\right] \tag{2.10}$$

where κ is the extensibility parameter $\kappa \in [0, 1]$. The constitutive equations of the Giesekus and PTT exponential viscoelastic model is reduced to Oldroyd-B model for α=0 and κ=0, respectively.

The numerical simulations of the viscoelastic problems is very challenging task and requires appropriate discretization methods. The task further gets difficult in case of the

pure polymer viscoelastic formulation i.e., neglecting the solvent viscosity and thus leading to the absence of the diffusion operator in the momentum equation. The resulting formulation shows several numerical challenges because of the absence of velocity coupling in the momentum equation. The absence of the diffusive operator from the momentum equation leads to severe difficulties in the numerical treatment of the three-field formulation in terms of discretization as well as solution techniques. Besides the usual velocity-pressure coupling, an additional stability condition i.e., the selection of approximating spaces for velocity and stress must be carefully considered. Moreover, restrictions regarding the applicability of widely used solution methods are faced, generally designing multigrid solvers using monolithic approaches impractical and preventing a reasonable implementation of decoupled/operator splitting approaches.
A possible remedy to this problem is to try to recover the velocity coupling back in the momentum equation. One way to retrieve the velocity coupling back into the momentum equation is by introducing the idea of change of variables, which was first introduced for the second-order fluid by Perera and Walters [48] and Mendelson et al. [43]. This change of variables is known as elastic viscous stress splitting (EVSS), which was later extended to the viscoelastic fluids by Baris et al. [9]. Recently, a novel technique has been introduced Westervoss [66], known as Tensor Stokes viscoelastic formulation. The Tensor Stokes formulation suggests a completely new way of splitting the extra stress tensor, and hence aids in the recovery of the diffusive type operator in the momentum equation. Such recovery of the diffusive type operator in the momentum equation is extremely helpful at the solver level. The EVSS and Tensor Stokes formulations are discussed in the next sections.

2.2.2 Elastic viscous stress splitting formulation

The splitting of the extra-stress tensor $\boldsymbol{S}$ into the elastic and viscous parts is called the elastic viscous stress splitting (EVSS) formulation [53]. The splitting is performed as follows,

$$\boldsymbol{S} = 2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E} \tag{2.11}$$

where, $\boldsymbol{E}$ is the elastic part of the extra stress tensor. Employing the splitting in the system of equations in (2.4) leads to the change of variables in the original viscoelastic formulation. The set of equations in the new variables is written as follows:

$$\rho\frac{\partial \mathbf{u}}{\partial t} - 2\eta_0 \nabla \cdot \mathbf{D}(\mathbf{u}) + \nabla p - \nabla \cdot \boldsymbol{E} = 0 \tag{2.12a}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{2.12b}$$

$$\lambda(2\eta_p \overset{\nabla}{\mathbf{D}}(\mathbf{u}) + \overset{\nabla}{\boldsymbol{E}}) + f\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}, \lambda\right)\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right) + h\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right) = 2\eta_p \mathbf{D}(\mathbf{u}) \tag{2.12c}$$

Equations (2.12a) and (2.12b) are Stokes equations coupled with an additional stress equation (2.12c) in the new variable $\boldsymbol{E}$. In the above mentioned formulation, $\mathbf{u}, p$ and $\boldsymbol{E}$ are the field variables that indicate the fluid's velocity, pressure and elastic part of the extra stress tensor, respectively. The constant η_0 is the sum of solvent viscosity η_s and polymer viscosity η_p, which is set equal to 1 for simplicity. The terms $\overset{\nabla}{\boldsymbol{E}}$ and $\overset{\nabla}{\mathbf{D}}(\mathbf{u})$ are upper-convected time derivative of the elastic part of the extra-stress tensor and deformation tensor, respectively.

$$\overset{\nabla}{\boldsymbol{E}} = \frac{\partial \boldsymbol{E}}{\partial t} + (\mathbf{u} \cdot \nabla)\boldsymbol{E} - \nabla \mathbf{u}^T \cdot \boldsymbol{E} - \boldsymbol{E} \cdot \nabla \mathbf{u} \tag{2.13}$$

$$\overset{\nabla}{\mathbf{D}} = \frac{\partial}{\partial t}\mathbf{D}(\mathbf{u}) + (\mathbf{u} \cdot \nabla)\mathbf{D}(\mathbf{u}) - \nabla \mathbf{u}^T \cdot \mathbf{D}(\mathbf{u}) - \mathbf{D}(\mathbf{u}) \cdot \nabla \mathbf{u} \tag{2.14}$$

Similar to the extra stress tensor formulation, the two functions $f\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}, \lambda\right)$ and $h\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right)$ in equation (2.12c) define the particular viscoelastic model. The constitutive equation for the simplest viscoelastic Oldroyd-B and UCM (Upper convected Maxwell) fluid model is obtained by setting $f\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}, \lambda\right)$=1 and $h\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right)$=0.

$$\lambda \overset{\nabla}{\boldsymbol{E}} + \boldsymbol{E} = -2\eta_p \lambda \overset{\nabla}{\mathbf{D}}(\mathbf{u}) \tag{2.15}$$

Giesekus model is achieved by setting $f\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}, \lambda\right)$=1 and $h\left(2\eta_p \mathbf{D}(\mathbf{u}) + \boldsymbol{E}\right) \neq 0$, one get.

$$\lambda \overset{\nabla}{\boldsymbol{E}} + \boldsymbol{E} + \frac{\alpha\lambda}{\eta_p}\left(\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right) \cdot \left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right)\right) = -2\eta_p \lambda \overset{\nabla}{\mathbf{D}}(\mathbf{u}) \tag{2.16}$$

In equation (2.12c), setting $f\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}, \lambda\right)$=*function* and $h\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right)$=0 we arrive at the PTT (Phan-Thien-Tanner) viscoelastic model.

$$\lambda(2\eta_p \overset{\nabla}{\mathbf{D}}(\mathbf{u}) + \overset{\nabla}{\boldsymbol{E}}) + g\left(\mathrm{tr}\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right)\right)\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right) = 2\eta_p \mathbf{D}(\mathbf{u}) \tag{2.17}$$

The stress function $g(\mathrm{tr}\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right))$ is either linear [51],

$$g\left(\mathrm{tr}\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right)\right) = 1 + \frac{\lambda\kappa}{\eta_p}\left(\mathrm{tr}\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right)\right) \tag{2.18}$$

or it takes the exponential form as proposed in [50].

$$g\left(\mathrm{tr}\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right)\right) = \exp\left[\frac{\lambda\kappa}{\eta_p}\left(\mathrm{tr}\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right)\right)\right] \tag{2.19}$$

In the upcoming subsection, we will introduce the Tensor-Stokes formulation.

2.2.3 Tensor Stokes formulation

The three field Tensor Stokes differential model for viscoelastic flow is introduced by Westervoss [66] obtained by decomposing the extra stress tensor $\boldsymbol{S}$ into the product of the non-symmetric viscosity tensor $\boldsymbol{T}$ and deformation tensor $\mathbf{D}(\mathbf{u})$,

$$\boldsymbol{S} = \boldsymbol{T} \cdot \mathbf{D}(\mathbf{u}) \tag{2.20}$$

By substituting the stress decomposition in the momentum equation (2.4a) a nonsymmetric tensor-valued viscosity is obtained in the momentum equation. An improved symmetric version of the decomposition is considered by considering the symmetric property of the extra-stress tensor;

$$\begin{aligned} \boldsymbol{S} &= \frac{1}{2}\left(\boldsymbol{S} + \boldsymbol{S}^T\right) \\ &= \frac{1}{2}\left(\boldsymbol{T} \cdot \mathbf{D}(\mathbf{u}) + \mathbf{D}(\mathbf{u}) \cdot \boldsymbol{T}^T\right) \end{aligned} \tag{2.21}$$

By substituting the stress decomposition in the set of equations (2.4) various potential benefits emerge from the numerical approach to simulate viscoelastic fluid flows that are discussed in *Chapter 3*. The three field Tensor Stokes viscoelastic formulation for the pure polymer melts reads as,

$$\rho\frac{\partial \mathbf{u}}{\partial t} - \frac{1}{2}\nabla \cdot \left(\boldsymbol{T} \cdot \mathbf{D}(\mathbf{u}) + \mathbf{D}(\mathbf{u}) \cdot \boldsymbol{T}^T\right) + \nabla p = 0 \tag{2.22a}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{2.22b}$$

$$\lambda(\overset{\nabla}{\boldsymbol{T} \cdot \mathbf{D}}) + f(\boldsymbol{T} \cdot \mathbf{D}, \lambda)(\boldsymbol{T} \cdot \mathbf{D}) + h(\boldsymbol{T} \cdot \mathbf{D}) = 2\eta_p \mathbf{D}(\mathbf{u}) \tag{2.22c}$$

Equations (2.22a) and (2.22b) are the modified Stokes equations coupled with the constitutive equation of a new kind (2.22c) in the variable $\boldsymbol{T}$. The term $(\overset{\nabla}{\boldsymbol{T} \cdot \mathbf{D}})$ is the upper-convected derivative, which reads as follows:

$$(\overset{\nabla}{\boldsymbol{T} \cdot \mathbf{D}}) = \frac{\partial}{\partial t}(\boldsymbol{T} \cdot \mathbf{D}) + (\mathbf{u} \cdot \nabla)(\boldsymbol{T} \cdot \mathbf{D}) - \left(\nabla \mathbf{u}^T \cdot (\boldsymbol{T} \cdot \mathbf{D}) + (\boldsymbol{T} \cdot \mathbf{D}) \cdot \nabla \mathbf{u}\right) \tag{2.23}$$

The two functions $f(\boldsymbol{T} \cdot \mathbf{D}, \lambda)$ and $h(\boldsymbol{T} \cdot \mathbf{D})$ in (2.22c) defines the particular viscoelastic fluid model. The constitutive equation for the simplest viscoelastic Oldroyd-B and UCM (Upper convected Maxwell) fluid model is obtained by setting $f(\boldsymbol{T} \cdot \mathbf{D}, \lambda)$=1 and $h(\boldsymbol{T} \cdot \mathbf{D})$=0.

$$\lambda(\overset{\nabla}{\boldsymbol{T} \cdot \mathbf{D}}) + \boldsymbol{T} \cdot \mathbf{D} = 2\eta_p \mathbf{D}(\mathbf{u}) \tag{2.24}$$

Giesekus model is acheived by setting $f(\boldsymbol{T} \cdot \mathbf{D}, \lambda)$=1 and $h(\boldsymbol{T} \cdot \mathbf{D}) \neq$0.

$$\lambda(\overset{\nabla}{\boldsymbol{T} \cdot \mathbf{D}}) + \boldsymbol{T} \cdot \mathbf{D} + \frac{\alpha\lambda}{\eta_p}((\boldsymbol{T} \cdot \mathbf{D}) \cdot (\boldsymbol{T} \cdot \mathbf{D})) = 2\eta_p \mathbf{D}(\mathbf{u}) \tag{2.25}$$

In (2.22c), setting $f(\boldsymbol{T}\cdot\mathbf{D},\lambda)$=*function* and $h(\boldsymbol{T}\cdot\mathbf{D})$=0 the PTT (Phan-Thien-Tanner) viscoelastic model is obtained, i.e.,

$$\overset{\nabla}{(\boldsymbol{T}\cdot\mathbf{D})}+g(\mathrm{tr}(\boldsymbol{T}\cdot\mathbf{D}))\,(\boldsymbol{T}\cdot\mathbf{D})=2\eta_p\mathbf{D}\,(\mathbf{u}) \tag{2.26}$$

The function $g(\mathrm{tr}\,(\boldsymbol{T}\cdot\mathbf{D}))$ is either linear [51]

$$g(\mathrm{tr}\,(\boldsymbol{T}\cdot\mathbf{D}))=1+\frac{\lambda\kappa}{\eta_p}\,(\mathrm{tr}\,(\boldsymbol{T}\cdot\mathbf{D})) \tag{2.27}$$

or it takes the exponential form as proposed by [50];

$$g(\mathrm{tr}\,(\boldsymbol{T}\cdot\mathbf{D}))=\exp\left[\frac{\lambda\kappa}{\eta_p}\,(\mathrm{tr}\,(\boldsymbol{T}\cdot\mathbf{D}))\right] \tag{2.28}$$

Having discussed various differential viscoelastic formulations for the fluids of interest, let us present the integral viscoelastic formulation in the next section.

2.3 Integral viscoelastic formulation

The constitutive equation used in the majority of the numerical studies of the viscoelastic flows is of the differential type. One of the main benefit of this formulation is that the constitutive equation for the stress depends only upon the current velocity and the stress fields. However, at the other side the multi-mode differential models are more expensive and difficult to solve in terms of increasing number of unknowns, numerical cost and runtime. Integral constitutive equations provide a feasible alternative approach where the multi modes do not directly result in an increase in the computational cost, at least from a numerical perspective. From a rheological standpoint, the integral viscoelastic fluid model is recommended because it takes into account the material's memory far more effectively than differential models do. The main feature of the integral viscoelastic formulation is the evaluation of the deformation history of the fluid flow by employing the integral equation for the stress. The time dependent Rivlin-Sawyers method [49, 35] for solving integral constitutive equation is given as follows:

$$\boldsymbol{S}\,(\mathbf{x},t)=\int_{-\infty}^{t}M\,(t-t')\,\mathbf{g}\,(\mathbf{B}_{t'}\,(\mathbf{x},t))\,dt' \tag{2.29}$$

In equation (2.29), $\boldsymbol{S}$ is the extra stress tensor, M is the memory function and $\mathbf{B}_{t'}\,(\mathbf{x},t)$ represents the finger strain tensor which is a field that measures the deformation of the fluid particle present at position $\mathbf{x}$, w.r.t. the reference time t', in the past. The specific

viscoelastic model is defined by the function $\mathbf{g}(\mathbf{B})$. The evolution equation for the Finger tensor is as follows:

$$\frac{\partial}{\partial s}\mathbf{B}_{t'}(s) + \mathbf{u}(s)\cdot\nabla\mathbf{B}_{t'}(s) - \nabla\mathbf{u}(s)^T\cdot\mathbf{B}_{t'}(s) - \mathbf{B}_{t'}(s)\cdot\nabla\mathbf{u}(s) = 0 \tag{2.30}$$

with $s\in[t',t]$ and initial condition $\mathbf{B}'_t(t') = \mathbf{I}$. The memory function M in equation (2.29) has the following form,

$$M(s) = \frac{\eta_p}{\lambda^2} exp\left(-\frac{s}{\lambda}\right) \tag{2.31}$$

with the polymeric viscosity η_p and relaxation parameter λ. The general form of the model function $\mathbf{g}(\mathbf{B})$ in equation (2.29) is given as

$$\mathbf{g}(\mathbf{B}) = \phi_1(I_1,I_2)\,\mathbf{B} + \phi_2(I_1,I_2)\,\mathbf{B}^{-1} \tag{2.32}$$

with I_1=tr($\mathbf{B}$) and I_2=$\frac{1}{2}\left(\mathrm{tr}(\mathbf{B})^2 - \mathrm{tr}(\mathbf{B}^2)\right)$ are the invariants of $\mathbf{B}$. In order to compute the stresses for 2D flows, we need to consider the 3D Finger tensor [59, 60] of the following form:

$$\mathbf{B}^{3D} = \begin{pmatrix} B_{11}^{3D} & B_{12}^{3D} & 0 \\ B_{12}^{3D} & B_{22}^{3D} & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{2.33}$$

To derive 2D integral model, calculate the invariants of $\mathbf{B}^{3D}$, which need to be inserted into the model function in equation (2.32).

$$\begin{aligned} I_1^{3D} &= \mathrm{tr}\left(\mathbf{B}^{3D}\right) = B_{11}^{3D} + B_{22}^{3D} + 1 \\ I_2^{3D} &= \frac{1}{2}\left[\mathrm{tr}\left(\mathbf{B}^{3D}\right)^2 - \mathrm{tr}\left(\left(\mathbf{B}^{3D}\right)^2\right)\right] \\ &= \frac{1}{2}\left[\left(B_{11}^{3D} + B_{22}^{3D} + 1\right)^2 - \left(\left(B_{11}^{3D}\right)^2 + 2\left(B_{12}^{3D}\right)^2 + \left(B_{22}^{3D}\right)^2 + 1\right)\right] \\ &= B_{11}^{3D}B_{22}^{3D} - \left(B_{12}^{3D}\right)^2 + B_{11}^{3D} + B_{22}^{3D} \\ &= \det\left(\mathbf{B}^{3D}\right) + \mathrm{tr}\left(\mathbf{B}^{3D}\right) - 1 \end{aligned} \tag{2.34}$$

With respect to 2D Finger tensor $\mathbf{B} = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$ equation (2.34) leads to,

$$I_1^{3D} = \mathrm{tr}(\mathbf{B}) + 1 = I_1 + 1, \quad I_2^{3D} = \mathrm{tr}(\mathbf{B}) + \det(\mathbf{B}) \tag{2.35}$$

but the det($\mathbf{B}$)=1 for incompressible flows, that is why $I_1^{3D} = I_2^{3D} = I_1 + 1$. The flow properties for certain integral viscoelastic models are obtained by specifying the

dampling functions ϕ_1 and ϕ_2 in equation (2.32). The 2D and 3D flow properties of integral Papanastasiou-Scriven-Macosko (PSM) [40] model are obtained by setting ϕ_1 and ϕ_2 as,

$$\begin{aligned} \phi_1^{3D} &= \frac{1}{1 + a\,(I_1 - 3) + b\,(I_1 - 3)}, \quad \phi_2^{3D} = 0 \\ \phi_1^{2D} &= \frac{1}{1 + \gamma\,(I_1 - 2)}, \quad \phi_2^{2D} = 0 \end{aligned} \tag{2.36}$$

The flow properties for the 3D Wagner model [40] for certain material parameters $f_1, n_1, f_2, n_2, \alpha$ and $I = \alpha I_1 + (1 - \alpha)\, I_2$ are,

$$\phi_1^{3D} = f_1 exp\left(-n_1\sqrt{I - 3}\right) + f_2 exp\left(-n_2\sqrt{I - 3}\right), \quad \phi_2^{3D} = 0 \tag{2.37}$$

The 2D Wagner model properties for $I = I_1 + 1$ are given as,

$$\phi_1^{2D} = f_1 exp\left(-n_1\sqrt{I_1 - 2}\right) + f_2 exp\left(-n_2\sqrt{I_1 - 2}\right), \quad \phi_2^{2D} = 0. \tag{2.38}$$

In the past, the integral constitutive equation flow calculations were based on the particle tracking and investigating the deformation history of the particle along its path [35]. Hulsen et al. [49] introduced a method different from the particle tracing called deformation field method (DFM). In DFM, the fields of Finger tensors are introduced and each field is tagged by the reference time of the deformation of the particle. In order to represent the deformation history accurately, the idea is to employ a reasonable number of fields and rationally disperse the reference times over the past history. The resulting fields are in accordance to the differential equation that solely depends only upon the current velocity and its gradients. The available discretization techniques for the differential viscoelastic models can be employed to solve the field equations. Out of particle tracing and deformation field method, DFM is the preferred method [31, 32]. Our focus is to formulate the stationary version of the integral viscoelastic fluid in order to compute the direct steady state solutions. Regarding the stationary integral viscoelastic model the momentum and continuity equation remains the same i.e.,

$$-2\eta_s \nabla \cdot \mathbf{D}\,(\mathbf{u}) - \nabla \cdot \boldsymbol{S} + \nabla p = 0 \tag{2.39a}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{2.39b}$$

However, the integral equation has to be reformulated carefully as the stress integral in equation (2.29) is time dependent. Also, even in the case of the stationary velocity field the Finger tensor (2.30) do not reach the steady state solution. Therefore, the stationary integral model requires the incorporation of an appropriate method of handling the accumulating time errors similar to the non-stationary deformation fields [49]. The quasi steady-state integral viscoelastic formulation reads as,

$$-2\eta_s \nabla \cdot \mathbf{D}(\mathbf{u}) - \nabla \cdot \boldsymbol{S} + \nabla p = 0 \tag{2.40a}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{2.40b}$$

$$\int_0^\infty M(s)\left[\phi_1(\mathrm{tr}(\mathbf{B}(s)))\mathbf{B}(s) + \phi_2(\mathrm{tr}(\mathbf{B}(s)))\mathbf{B}^{-1}(s)\right]ds = \boldsymbol{S} \tag{2.40c}$$

coupled with the evolution equation,

$$\frac{\partial}{\partial s}\mathbf{B}(s) + \mathbf{u}\cdot\nabla\mathbf{B}(s) - \nabla\mathbf{u}^T\cdot\mathbf{B}(s) - \mathbf{B}(s)\cdot\nabla\mathbf{u} = 0 \tag{2.41}$$

for $s \in [0, \mathrm{inf})$ together with the initial condition $\mathbf{B}(0) = \mathbf{I}$.

Within the context of fully developed channel flows, viscoelastic fluid flow models with integral constitutive equations are taken into account to establish the corresponding flow profiles identical to those in the differential case. The flow properties for the fully developed one-dimensional channel flows are,

$$\mathbf{u} = \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} u(y) \\ 0 \end{pmatrix},\ \frac{\partial u}{\partial x} = \frac{\partial v}{\partial x} = \frac{\partial v}{\partial y} = 0, \frac{\partial S_{ij}}{\partial x} = \frac{\partial B_{ij}}{\partial x} = 0\ for\ i,j \in \{1,2\} \tag{2.42}$$

Using the flow properties in equation (2.42), one-dimensional set of equations for the evolution equation of the Finger Tensor $\mathbf{B}$ (2.41) are obtained as follows:

$$\frac{\partial}{\partial s}B_{11}(s) = 2B_{12}(s)\frac{\partial u}{\partial y},\ \frac{\partial}{\partial s}B_{12}(s) = \frac{\partial u}{\partial y}B_{22}(s),\ \frac{\partial}{\partial s}B_{22}(s) = 0 \tag{2.43}$$

for $s \in [0, \mathrm{inf})$. Subject to the initial conditions $\mathbf{B}(0) = \mathbf{I}$ the analytic expressions for the components of the Finger tensor reads as,

$$B_{22}(s) = 1,\ B_{12}(s) = s\frac{\partial u}{\partial y},\ B_{11}(s) = s^2\left(\frac{\partial u}{\partial y}\right)^2 + 1 \tag{2.44}$$

Using the components of $\mathbf{B}$, it's inverse can be computed analytically [12]. Insert (2.44) in the integral equation for the stress (2.40c) in the stationary integral model the set of equations are given as,

$$\begin{aligned} -2\eta_s \nabla \cdot \mathbf{D}(\mathbf{u}) - \nabla \cdot \boldsymbol{S} + \nabla p = 0 \\ \nabla \cdot \mathbf{u} = 0 \end{aligned} \tag{2.45}$$

$$\begin{aligned} \boldsymbol{S} = \int_0^\infty M(s)\Big[\phi_1(\mathrm{tr}(\mathbf{B}))\begin{pmatrix} s^2\left(\frac{\partial u}{\partial y}\right)^2 + 1 & s\frac{\partial u}{\partial y} \\ s\frac{\partial u}{\partial y} & 1 \end{pmatrix} ... \\ ... + \phi_2(\mathrm{tr}(\mathbf{B}))\begin{pmatrix} 1 & -s\frac{\partial u}{\partial y} \\ -s\frac{\partial u}{\partial y} & s^2\left(\frac{\partial u}{\partial y}\right)^2 + 1 \end{pmatrix}\Big]ds \end{aligned} \tag{2.46}$$

where the trace of the finger tensor is $\mathrm{tr}(\mathbf{B}) = B_{11} + B_{22} = s^2\left(\frac{\partial u}{\partial y}\right)^2 + 2$.
In equation (2.46) the stress tensor is dependent upon $\frac{\partial u}{\partial y}$ therefore, the need to treat the stress variable independently vanishes. Additionally, the fully developed flow profiles are in principle determined by momentum equation according to the differential formulation, which provides a relation between the velocity and pressure drop over the channel length. Substitute (2.46) into the momentum equation in (2.45) results in the fully developed velocity profile in terms of the integral viscoelastic model. The three field integral viscoelastic model is brought down to a non linear two field problem in the unknowns $(\frac{\partial u}{\partial y}, \frac{\partial p}{\partial x})$. The stress profiles (2.46) are computed only in the post processing manner. Substitute the properties of the Wagner model (2.38) in the constitutive equation (2.46) the integral equation for the Wagner model reads as,

$$\begin{aligned} \boldsymbol{S} = \frac{\eta_p}{\lambda^2}\int_0^\infty exp(\frac{-s}{\lambda})\Bigg(\Bigg[f exp\left(-n_1\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right) + ... \\ ... (1-f)\, exp\left(-n_2\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right)\Bigg]\begin{pmatrix} s^2\left(\frac{\partial u}{\partial y}\right)^2 + 1 & s\frac{\partial u}{\partial y} \\ s\frac{\partial u}{\partial y} & 1 \end{pmatrix}\Bigg) ds \end{aligned} \tag{2.47}$$

Perform integration by parts in (2.47) to get the components of the tensor $\boldsymbol{S}$

$$\begin{aligned} S_{11} &= 2\frac{\eta_p}{\lambda^2}\left[f\left(\frac{1}{\lambda} + n_1\sqrt{\left(\frac{\partial u}{\partial y}\right)^2}\right)^{-3} + (1-f)\left(\frac{1}{\lambda} + n_2\sqrt{\left(\frac{\partial u}{\partial y}\right)^2}\right)^{-3}\right]\left(\frac{\partial u}{\partial y}\right)^2 \\ S_{12} &= \frac{\eta_p}{\lambda^2}\left[f\left(\frac{1}{\lambda} + n_1\sqrt{\left(\frac{\partial u}{\partial y}\right)^2}\right)^{-2} + (1-f)\left(\frac{1}{\lambda} + n_2\sqrt{\left(\frac{\partial u}{\partial y}\right)^2}\right)^{-2}\right]\frac{\partial u}{\partial y} \\ S_{21} &= \frac{\eta_p}{\lambda^2}\left[f\left(\frac{1}{\lambda} + n_1\sqrt{\left(\frac{\partial u}{\partial y}\right)^2}\right)^{-2} + (1-f)\left(\frac{1}{\lambda} + n_2\sqrt{\left(\frac{\partial u}{\partial y}\right)^2}\right)^{-2}\right]\frac{\partial u}{\partial y} \\ S_{22} &= 0 \end{aligned} \tag{2.48}$$

The algorithm for the fully developed 2-D channel flows for the Wagner integral viscoelastic model is given as follows:
Let $(\mathbf{u}^n, p^n)$ be the known approximation after n steps. The step $(n+1)$ of the algorithm consists of first computing $(\mathbf{u}^{n+1}, p^{n+1})$ by solving the Stokes equation with the zero RHS and in the second step compute element wise constant $\boldsymbol{S}^{n+1}$, with the new velocity $\mathbf{u}^{n+1}$. The iteration $(n+1)$ consists of first computing $(\mathbf{u}^{n+1}, p^{n+1})$ as follows,

$$\begin{aligned} -2\eta_s \nabla\cdot\boldsymbol{D}\left(\mathbf{u}^{n+1}\right) + \nabla p^{n+1} &= \nabla\cdot\boldsymbol{S}^n \\ \nabla\cdot\mathbf{u}^{n+1} &= 0 \end{aligned} \tag{2.49}$$

and in the second step compute $\boldsymbol{S}^{n+1}$:

$$\begin{aligned}\boldsymbol{S}^{n+1} = \frac{\eta_p}{\lambda^2}\int_0^{\infty} exp(\frac{-s}{\lambda})\Bigg(\Bigg[fexp\left(-n_1\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right) + ... \\ ...\,(1-f)\,exp\left(-n_2\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right)\Bigg]\begin{pmatrix} s^2\left(\frac{\partial u}{\partial y}\right)^2+1 & s\frac{\partial u}{\partial y} \\ s\frac{\partial u}{\partial y} & 1\end{pmatrix}\Bigg)ds\end{aligned} \tag{2.50}$$

whose components are given in (2.48).

Having discussed the two-field integral viscoelastic formulation with extra-stress tensor computed in the post-processing step, let us present the viscoelastic formulations for the pure polymer melts in the following sections.

2.3.1 Elastic viscous stress splitting integral formulation

While considering the pure polymer melts in the above formulation the solvent viscosity vanishes and hence the velocity coupling in the momentum equation. To regain the velocity coupling into the momentum equation, the splitting of the extra-stress tensor as in the differential case is performed. Such splitting results in the EVSS and Tensor Stokes integral viscoelastic formulation with zero solvent viscosity i.e., $\eta_s = 0$. The EVSS splitting [53] is given as:

$$\boldsymbol{S} = \boldsymbol{E} + 2\eta_p \mathbf{D}\left(\mathbf{u}\right) \tag{2.51}$$

Consider the constitutive equation (2.47) for the Wagner model.

$$\begin{aligned}\boldsymbol{S} = \frac{\eta_p}{\lambda^2}\int_0^{\infty} exp(\frac{-s}{\lambda})\Bigg(\Bigg[fexp\left(-n_1\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right) + ... \\ ...\,(1-f)\,exp\left(-n_2\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right)\Bigg]\Bigg\{\begin{pmatrix} s^2\left(\frac{\partial u}{\partial y}\right)^2 & s\frac{\partial u}{\partial y} \\ s\frac{\partial u}{\partial y} & 0\end{pmatrix}+\mathbf{I}\Bigg\}\Bigg)ds\end{aligned} \tag{2.52}$$

Separate the isotropic part given by the identity $\mathbf{I}$ a diagonal matrix $\mathbf{F} \in \mathbb{R}^{2\times 2}$ is obtained,

$$\begin{aligned}\mathbf{F} = \frac{\eta_p}{\lambda^2}\int_0^{\infty} exp(\frac{-s}{\lambda})\Bigg(\Bigg[fexp\left(-n_1\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right) + ... \\ ...\,(1-f)\,exp\left(-n_2\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right)\Bigg]\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}\Bigg)ds\end{aligned} \tag{2.53}$$

whose components are,

$$\begin{aligned}
\chi_{11} = \frac{\eta_p}{\lambda^2}\int_0^\infty exp(\frac{-s}{\lambda})\Bigg[fexp\left(-n_1\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right) + ... \\
... (1-f)\, exp\left(-n_2\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right)\Bigg] ds \\
\chi_{22} = \frac{\eta_p}{\lambda^2}\int_0^\infty exp(\frac{-s}{\lambda})\Bigg[fexp\left(-n_1\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right) + ... \\
... (1-f)\, exp\left(-n_2\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right)\Bigg] ds \\
\chi_{12} = \chi_{21} = 0
\end{aligned} \tag{2.54}$$

The two field EVSS integral formulation for the pure polymer melts ($\eta_s = 0$) reads as follows:

$$\begin{aligned}
-2\eta_0 \nabla \cdot \mathbf{D}(\mathbf{u}) - \nabla \cdot \mathbf{F} - \nabla \cdot \boldsymbol{E} + \nabla p = 0 \\
\nabla \cdot \mathbf{u} = 0
\end{aligned} \tag{2.55}$$

where $\eta_0 = \eta_s + \eta_p$=1 and η_s=0. In equation (2.54) the components depends upon y and $\partial u/\partial y$, the term $\nabla \cdot \mathbf{F}$ in (2.55) can be written as $\nabla\chi$. Hence the momentum equation is reconsidered having a modified pressure i.e. $P = p - \chi$ instead of p. The isotropic part $\mathbf{F}$ does not affect the velocity but only the pressure field and extra-stress tensor. Also, the extra-stress is of least interest in the current scenario since they are computed only in the post processing fashion. The change in pressure is not considered here because the primary goal of the current formulation is the assessment of the tensor $\boldsymbol{E}$, which is computed using the velocity field. Thus, the original EVSS formulation is considered with the tensor $\boldsymbol{E}$ computed as follows,

$$\begin{aligned}
\boldsymbol{E} = \frac{\eta_p}{\lambda^2}\int_0^\infty exp(\frac{-s}{\lambda})\Bigg(\Bigg[fexp\left(-n_1\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right) + (1-f)\, ... \\
...exp\left(-n_2\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right)\Bigg]\left(\begin{pmatrix} s^2\left(\frac{\partial u}{\partial y}\right)^2 & s\frac{\partial u}{\partial y} \\ s\frac{\partial u}{\partial y} & 0 \end{pmatrix} - \begin{pmatrix} 0 & s\frac{\partial u}{\partial y} \\ s\frac{\partial u}{\partial y} & 0 \end{pmatrix}\right) ds
\end{aligned} \tag{2.56}$$

Integration by parts results in the following components of tensor $\boldsymbol{E}$,

$$
\begin{aligned}
E_{11} &= 2\frac{\eta_p}{\lambda^2}\left[f\left(\frac{1}{\lambda}+n_1\sqrt{\left(\frac{\partial u}{\partial y}\right)^2}\right)^{-3}+(1-f)\left(\frac{1}{\lambda}+n_2\sqrt{\left(\frac{\partial u}{\partial y}\right)^2}\right)^{-3}\right]\left(\frac{\partial u}{\partial y}\right)^2\\
E_{12} &= \eta_p\frac{\partial u}{\partial y}\left[\frac{f}{\lambda^2}\left(\frac{1}{\lambda}+n_1\sqrt{\left(\frac{\partial u}{\partial y}\right)^2}\right)^{-2}+\frac{(1-f)}{\lambda^2}\left(\frac{1}{\lambda}+n_2\sqrt{\left(\frac{\partial u}{\partial y}\right)^2}\right)^{-2}-1\right]\\
E_{21} &= \eta_p\frac{\partial u}{\partial y}\left[\frac{f}{\lambda^2}\left(\frac{1}{\lambda}+n_1\sqrt{\left(\frac{\partial u}{\partial y}\right)^2}\right)^{-2}+\frac{(1-f)}{\lambda^2}\left(\frac{1}{\lambda}+n_2\sqrt{\left(\frac{\partial u}{\partial y}\right)^2}\right)^{-2}-1\right]\\
E_{22} &= 0
\end{aligned}
\tag{2.57}
$$

The EVSS formulation for the integral models is quite trivial i.e., the tensor $\boldsymbol{E}$ can be obtained by simply subtracting the deformation tensor from the extra stress tensor. Having known the integral constitutive equation for the stress in EVSS formulation one can develop the simple two field algorithm to obtain the results.

The problem for the iterative approach is defined as follows:
Let $(\mathbf{u}^n, p^n)$ be the known approximation after n steps. The step $(n+1)$ of the algorithm consists of first computing $(\mathbf{u}^{n+1}, p^{n+1})$ by solving the stokes equation with the zero RHS and in the second step compute element wise constant $\boldsymbol{E}^{n+1}$, with the new velocity $\mathbf{u}^{n+1}$. The iteration $(n+1)$ consists of first computing $(\mathbf{u}^{n+1}, p^{n+1})$ as follows:

$$
\begin{aligned}
-2\eta_0\nabla\cdot\boldsymbol{D}\left(\mathbf{u}^{n+1}\right)+\nabla p^{n+1} &= \nabla\cdot\boldsymbol{E}^n\\
\nabla\cdot\mathbf{u}^{n+1} &= 0
\end{aligned}
\tag{2.58}
$$

and in the second step we compute $\boldsymbol{E}^{n+1}$,

$$
\begin{aligned}
\boldsymbol{E}^{n+1} = \frac{\eta_p}{\lambda^2}\int_0^\infty exp(\frac{-s}{\lambda})\Bigg(\Bigg[fexp\left(-n_1\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right)+(1-f)\,...\\
...exp\left(-n_2\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right)\Bigg]\left(s^2\begin{pmatrix}\left(\frac{\partial u}{\partial y}\right)^2 & s\frac{\partial u}{\partial y}\\ s\frac{\partial u}{\partial y} & 0\end{pmatrix}-\begin{pmatrix}0 & s\frac{\partial u}{\partial y}\\ s\frac{\partial u}{\partial y} & 0\end{pmatrix}\right)ds
\end{aligned}
\tag{2.59}
$$

whose components are given in equation (2.57). Let us briefly explain the integral formulation in terms of Tensor Stokes formulation.

2.3.2 Tensor Stokes integral viscoelastic formulation

Tensor Stokes formulation suggests the splitting of the extra stress tensor as follows,

$$\boldsymbol{S} = \boldsymbol{T} \cdot \mathbf{D}(\mathbf{u}) \tag{2.60}$$

Decomposition of the extra-stress tensor given in equation (2.47) for the Wagner model results in

$$\begin{aligned} \boldsymbol{S} = \frac{\eta_p}{\lambda^2} \int_0^\infty exp(\frac{-s}{\lambda}) \Bigg[f exp\left(-n_1 \sqrt{s^2 \left(\frac{\partial u}{\partial y} \right)^2} \right) + ... \\ ... (1-f) exp\left(-n_2 \sqrt{s^2 \left(\frac{\partial u}{\partial y} \right)^2} \right) \Bigg] \begin{pmatrix} s & s^2 \frac{\partial u}{\partial y} \\ 0 & s \end{pmatrix} ds \quad \mathbf{D}(\mathbf{u}) + \mathbf{F} \end{aligned} \tag{2.61}$$

where the integral $\mathbf{F}$ is the isotropic part given in equation (2.53).

The integral form of the tensor $\boldsymbol{T}$ is,

$$\begin{aligned} \boldsymbol{T} = 2\frac{\eta_p}{\lambda^2} \int_0^\infty exp(\frac{-s}{\lambda}) \Bigg[f exp\left(-n_1 \sqrt{s^2 \left(\frac{\partial u}{\partial y} \right)^2} \right) + ... \\ ... (1-f) exp\left(-n_2 \sqrt{s^2 \left(\frac{\partial u}{\partial y} \right)^2} \right) \Bigg] \begin{pmatrix} s & s^2 \frac{\partial u}{\partial y} \\ 0 & s \end{pmatrix} ds \end{aligned} \tag{2.62}$$

Analytic expressions for the components of the tensor $\boldsymbol{T}$ are,

$$\begin{aligned} T_{11} &= 2\frac{\eta_p}{\lambda^2} \left[f \left(\frac{1}{\lambda} + n_1 \sqrt{\left(\frac{\partial u}{\partial y} \right)^2} \right)^{-2} + (1-f) \left(\frac{1}{\lambda} + n_2 \sqrt{\left(\frac{\partial u}{\partial y} \right)^2} \right)^{-2} \right] \\ T_{12} &= 4\frac{\eta_p}{\lambda^2} \left[f \left(\frac{1}{\lambda} + n_1 \sqrt{\left(\frac{\partial u}{\partial y} \right)^2} \right)^{-3} + (1-f) \left(\frac{1}{\lambda} + n_2 \sqrt{\left(\frac{\partial u}{\partial y} \right)^2} \right)^{-3} \right] \frac{\partial u}{\partial y} \\ T_{21} &= 0 \\ T_{22} &= 2\frac{\eta_p}{\lambda^2} \left[f \left(\frac{1}{\lambda} + n_1 \sqrt{\left(\frac{\partial u}{\partial y} \right)^2} \right)^{-2} + (1-f) \left(\frac{1}{\lambda} + n_2 \sqrt{\left(\frac{\partial u}{\partial y} \right)^2} \right)^{-2} \right] \end{aligned} \tag{2.63}$$

The problem for the two field Tensor Stokes Wagner viscoelastic formulation reads as: Let $(\mathbf{u}^n, p^n)$ be the known approximation after n steps. The step $(n+1)$ of the algorithm consists of first computing $(\mathbf{u}^{n+1}, p^{n+1})$ by solving the stokes equation using a constant $\boldsymbol{T}$ value and then computing element wise constant $\boldsymbol{T}^{n+1}$, with the new velocity $\mathbf{u}^{n+1}$. The iteration $(n+1)$ consists of first computing $(\mathbf{u}^{n+1}, p^{n+1})$ as follows,

$$\begin{aligned} -\nabla \cdot \frac{1}{2} \left(\boldsymbol{T}(\mathbf{u}^n) \mathbf{D}\left(\mathbf{u}^{n+1}\right) + \mathbf{D}\left(\mathbf{u}^{n+1}\right) \boldsymbol{T}^T(\mathbf{u}^n) \right) + \nabla p^{n+1} = 0 \\ \nabla \cdot \mathbf{u}^{n+1} = 0 \end{aligned} \tag{2.64}$$

and in the second step we compute $\boldsymbol{T}^{n+1}$

$$\boldsymbol{T}^{n+1} = 2\frac{\eta_p}{\lambda^2}\int_0^\infty exp(\frac{-s}{\lambda})\left[fexp\left(-n_1\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right) + ... \right.$$
$$\left. ... (1-f)\, exp\left(-n_2\sqrt{s^2\left(\frac{\partial u}{\partial y}\right)^2}\right)\right]\begin{pmatrix} s & s^2\frac{\partial u}{\partial y} \\ 0 & s \end{pmatrix} ds \tag{2.65}$$

whose components are given in equation (2.63). The material parameters for the Wagner model are $f = 0.57$, $n_1 = 0.31$ and $n_2 = 0.106$ as shown by [40]. The Wagner model employed in Tensor Stokes and EVSS formulation in two-field formulation depicts the viscoelastic effects inside of a channel as it is capable of producing the shear thinning effect.
Having discussed the differential and integral viscoelastic formulations, let us briefly discuss the boundary conditions in the upcoming section.

2.4 Boundary conditions

The flow phenomenons occurring in the real world are not physically bounded or closed e.g. the air flow around a car or an aeroplane. However, it is essential to provide the boundary conditions when the physical phenomenon is mathematically treated as a set of partial differential equations because otherwise, the set of equations cannot be solved. Following is a discussion of the typical boundary conditions employed for the incompressible flows:

- *Dirichlet boundary conditions:*
  The values of the function that the solution takes on the boundary are defined by Dirichlet boundary conditions. The velocity field is specified on either the entire domain or a portion of the boundary for the incompressible Stokes flow.

$$\mathbf{u}(\mathbf{x}) = \mathbf{g}(\mathbf{x}) \quad \text{on} \quad \Gamma_D \tag{2.66}$$

  where Γ_D specifies Dirichlet part of the boundary $\partial\Omega$. The special case of $\mathbf{g}(\mathbf{x})$=0 results in no-slip boundary conditions.

$$\mathbf{u}(\mathbf{x}) = 0 \quad \text{on} \quad \Gamma_D \tag{2.67}$$

- *Do nothing boundary conditions:* The do-nothing boundary condition leaves the solution free on that part of the boundary. Do-nothing boundary condition is obtained by the integration by parts of the weak formulation of the momentum equation that results in,

$$p\mathbf{n} - 2\eta_s \mathbf{D}(\mathbf{u}) \cdot \mathbf{n} - \boldsymbol{S} \cdot \mathbf{n} = 0 \quad \text{on} \quad \Gamma_N \tag{2.68}$$

where Γ_N specifies Neumann part of the boundary $\partial\Omega$. If no other outflow boundary condition is available, do-nothing boundary condition is frequently utilized. From the mathematical perspective, do-nothing boundary conditions are referred as natural boundary conditions.

- *Velocity and Stress boundary conditions:* For the considered problem, the computational domain is represented by Ω. In our numerical computations, the inflow and the outflow part of the boundary are denoted as $\Gamma_{in}, \Gamma_{out}$, respectively. The usual boundary condition at the inflow is a prescribed velocity profile, $\mathbf{u}(y)$. At the outflow, one can either prescribe the Dirichlet data similar to the inflow or prescribe some do-nothing boundary conditions. The no-slip condition is enforced at solid boundaries for the velocity field. Consider the velocity profile for the Poiseuille flow:

$$\mathbf{u}(y) = \begin{pmatrix} U(1-y^2) \\ 0 \end{pmatrix}. \tag{2.69}$$

The inflow boundary conditions for the constitutive equation for stress $(\boldsymbol{S}, \boldsymbol{E}, \boldsymbol{T})$ is computed on the basis of the velocity function $\mathbf{u}(y)$ [68]. For the simplest Oldroyd-B fluid, the stress boundary condition for the standard viscoelastic formulation obtained from equation (2.69) is given as:

$$\boldsymbol{S} = \begin{pmatrix} 8\lambda\eta_p U^2 y^2 & -2\eta_p U y \\ -2\eta_p U y & 0 \end{pmatrix}. \tag{2.70}$$

The stress boundary condition for the EVSS variable $\boldsymbol{E}$ and Tensor Stokes variable $\boldsymbol{T}$ can be obtained by employing the stress splitting

$$\boldsymbol{S} = \boldsymbol{E} + 2\eta_p \mathbf{D}(\mathbf{u}) \tag{2.71}$$

and

$$\boldsymbol{S} = \boldsymbol{T} \cdot \mathbf{D}(\mathbf{u}), \tag{2.72}$$

respectively.

At the outflow, natural boundary conditions are considered because viscoelastic fluids have a memory [29]. A thorough investigation of the boundary conditions is conducted in Chapter 7 and 8, focusing on the in- and outflow profiles.

In the upcoming chapter, we will outline the numerical challenges related to the simulation of viscoelastic flow problems.

3

Numerical challenges posed by viscoelastic fluids

The solution of the pure polymer melts requires the solution of the Stokes problem with zero solvent viscosity coupled with a constitutive law expressing the relationship between the stress tensor and velocity gradient. To obtain such solution is not an easy task, and numerical simulations of such viscoelastic flows is very challenging. Therefore, the development of stable and efficient numerical methods for computing such fluid flows is extremely important. For pure polymer melts, the corresponding diffusive operator vanishes from the momentum equation of the Stokes problem, leading to a number of issues with the underlying three-field formulation discretization and solution strategies. Regarding the discretization of the three field formulation for the pure polmyer melts, in addition to the usual velocity-pressure coupling, there is another stability condition that must be taken into consideration while choosing the approximation spaces with respect to the velocity and stress variables. An inf-sup condition must be met before choosing the finite element spaces used to estimate the velocity, pressure, and extra-stress field. The detailed discussion regarding the discretization techniques is followed in *Chapter 4*. The three field formulation can be treated in a monolithic or decoupled fashion. For the pure polymer melts, limitations are observed for both the solution strategies, for the monolithic approach the application of the multigrid solvers become quite unpractical and the solution via the decoupled approach is impossible to obtain without any additional stabilization. In this thesis, we have tried to overcome the practical issues that arise while dealing with the pure polymer melts. The next two sections address the viscoelastic formulations namely, Elastic viscous stress splitting (EVSS) and Tensor Stokes formulation obtained by employing the change of variable in the standard viscoelastic formulation with the aim to recover the diffusive operator in the momentum equation of the Stokes problem.

3.1 Elastic viscous stress splitting formulation

Elastic Viscous Stress Splitting (EVSS) is a fundamental method for solving viscoleastic flow problems in the absence of the solvent viscosty i.e., $\eta_s = 0$. The EVSS formulation is obtained by performing the change of variables in the standard viscoelastic formulation given in (2.4). This change of variables recovers the velocity coupling back into the momentum equation by the introduction of the diffusive operator into the momentum equation. Let us reconsider the EVSS formulation (2.12),

$$\begin{aligned} \rho\frac{\partial \mathbf{u}}{\partial t} - 2\eta_0 \nabla\cdot\mathbf{D}(\mathbf{u}) + \nabla p - \nabla\cdot\boldsymbol{E} &= 0 \\ \nabla\cdot\mathbf{u} &= 0 \\ \lambda(2\eta_p \overset{\nabla}{\mathbf{D}}(\mathbf{u}) + \overset{\nabla}{\boldsymbol{E}}) + f\left(2\eta_p\mathbf{D}(\mathbf{u}) + \mathbf{E}, \lambda\right)\left(2\eta_p\mathbf{D}(\mathbf{u}) + \mathbf{E}\right) + & \\ h\left(2\eta_p\mathbf{D}(\mathbf{u}) + \mathbf{E}\right) &= 2\eta_p\mathbf{D}(\mathbf{u}) \end{aligned} \tag{3.1}$$

where $\overset{\nabla}{\boldsymbol{E}}$ and $\overset{\nabla}{\mathbf{D}}$ are given as,

$$\overset{\nabla}{\boldsymbol{E}} = \frac{\partial \boldsymbol{E}}{\partial t} + (\mathbf{u}\cdot\nabla)\boldsymbol{E} - \nabla\mathbf{u}^T\cdot\boldsymbol{E} - \boldsymbol{E}\cdot\nabla\mathbf{u} \tag{3.2}$$

$$\overset{\nabla}{\mathbf{D}} = \frac{\partial}{\partial t}\mathbf{D}(\mathbf{u}) + (\mathbf{u}\cdot\nabla)\mathbf{D}(\mathbf{u}) - \nabla\mathbf{u}^T\cdot\mathbf{D}(\mathbf{u}) - \mathbf{D}(\mathbf{u})\cdot\nabla\mathbf{u} \tag{3.3}$$

The deformation tensor is now scaled by the total viscosity $\eta_0 = \eta_s + \eta_p$ instead of just scaled by the solvent component η_s. The momentum equation contains the diffusive operator even though the contribution from the solvent viscosity is zero.

There are also certain drawbacks of this formulation as EVSS consists of elastic part of extra stress as primal stress variable instead of the extra-stress tensor. The appearance of additional terms in the upper-convected derivative of the strain-rate tensor $\mathbf{D}$ in the constitutive equation for stress required special attention. Second order velocity derivatives which are very challenging to handle computationally are involved in the upper-convected derivative of the rate of deformation tensor. The typical method for handling these higher order derivatives is to increase the problem size to a four-field formulation by treating the strain-rate tensor as an additional variable thus avoiding the higher-order derivatives in the weak formulation. Rajagopalan et al. [53] introduces the rate of deformation tensor as an additional variable,

$$\boldsymbol{R} = \mathbf{D}(\mathbf{u}) \tag{3.4}$$

The addition of the deformation tensor as an additional unknown leads to the four field formulation in $\mathbf{u}, p, \boldsymbol{E}$ and $\boldsymbol{R}$.

$$\begin{aligned} -2\eta_0 \nabla \cdot \mathbf{D}(\mathbf{u}) + \nabla p - \nabla \cdot \boldsymbol{E} &= 0 \\ \nabla \cdot \mathbf{u} &= 0 \\ \lambda(2\eta_p \overset{\nabla}{\mathbf{D}}(\mathbf{u}) + \overset{\nabla}{\boldsymbol{E}}) + f\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}, \lambda\right)\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right) + & \\ h\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right) &= 2\eta_p \mathbf{D}(\mathbf{u}) \\ \boldsymbol{R} &= \mathbf{D}(\mathbf{u}) \end{aligned} \tag{3.5}$$

Rajagopalan et al. [53] were quite successful while working with the four field EVSS formulation as the method helped them to compute wide variety of flows. Guenette et al. [28] suggested a discrete variant of the EVSS approach called DEVSS. The original EVSS formulation has been modified to create the DEVSS. The four field formulation is made possible in DEVSS by the introduction of the rate of the deformation tensor as an additional variable. The DEVSS does not execute an explicit variable change, but the momentum equation does include a regularizing diffusion term. Also, there are no second order velocity derivatives in the constitutive equation because no variable changes are made in the DEVSS. The rate of deformation tensor is added to the momentum equation left and right hand side in the DEVSS version. The stationary set of equations reads as,

$$\begin{aligned} -2\eta_s \nabla \cdot \mathbf{D}(\mathbf{u}) - 2\alpha \nabla \cdot \mathbf{D}(\mathbf{u}) + \nabla p &= \nabla \cdot \boldsymbol{S} - 2\alpha \nabla \cdot \boldsymbol{\nu} \\ \nabla \cdot \mathbf{u} &= 0 \\ \lambda \overset{\nabla}{\boldsymbol{S}} + f(\boldsymbol{S}, \lambda)\, \boldsymbol{S} + h(\boldsymbol{S}) &= 2\eta_p \mathbf{D}(\mathbf{u}) \\ \boldsymbol{\nu} &= \mathbf{D}(\mathbf{u}) \end{aligned} \tag{3.6}$$

where $\alpha > 0$ is an artificial model parameter. Considering DEVSS on the continuous level, it can be seen that the set of equations of the classic viscoelastic formulation (2.4) remains unchanged. Due to the addition of a second variable, $\boldsymbol{\nu}$, which helps to stabilise the formulation, there may be some approximation errors when the formulation is taken into account on the discrete level. The formulation proposed by Guenette et al. [28] shows good numerical properties. However, the problem size is again increased to the four field formulation $\mathbf{u}, p, \boldsymbol{S}$ and $\boldsymbol{\nu}$ which inturn increases the computational effort. Guenette et al. [28] employed a decoupled method (3.7)-(3.9) to obtain the results for the four field formulation which is given as follows:
At the first step solve for the Stokes problem $\mathbf{u}^{n+1}, p^{n+1}$,

$$\begin{aligned} -2\left(\eta_s + \alpha\right) \nabla \cdot \mathbf{D}\left(\mathbf{u}^{n+1}\right) + \nabla p^{n+1} &= \nabla \cdot \left(\boldsymbol{S}^n - 2\alpha \boldsymbol{\nu}^n\right) \\ \nabla \cdot \mathbf{u}^{n+1} &= 0 \end{aligned} \tag{3.7}$$

In the second step solve for $\boldsymbol{S}^{n+1}$

$$\lambda \overset{\nabla}{\boldsymbol{S}}^{n+1} + f(\boldsymbol{S}^n, \lambda)\, \boldsymbol{S}^{n+1} + h\left(\boldsymbol{S}^{n+1}\right) = 2\eta_p \mathbf{D}\left(\mathbf{u}^{n+1}\right) \tag{3.8}$$

In the third step solve for $\boldsymbol{\nu}^{n+1}$

$$\boldsymbol{\nu}^{n+1} = \mathbf{D}\left(\mathbf{u}^{n+1}\right) \tag{3.9}$$

One advantage of employing the DEVSS is the capability to address the stationary differential viscoelastic flow model in a decoupled manner, even when the solvent viscosity is absent. This is specially beneficial for simulating viscoelastic flows in the absence of solvent viscosity where multiple modes and thus the extra stress tensors are taken into account simultaneously, which definitely makes application of multigrid solver unrealistic. However, the increase in the problem size requires high computational cost. The benefit of this formulation is that it allows for the decoupled solution of viscoelastic flows, even in the condition of zero solvent viscosity, without requiring a change in variables. In order to obtain solutions of the viscoelastic fluids with zero solvent viscosity, the four field EVSS formulation or its discrete version DEVSS appears to be fairly helpful. The additional computational cost is caused by the addition of a new variable, rate for the deformation tensor.

Our goal is to benefit from the EVSS formulation without increasing the problem size. In this regard, one could transform the convective term $(\mathbf{u}\cdot\nabla)\,\mathbf{D}(\mathbf{u})$ in equation (3.3) by taking into account the incompressibility constraint i.e., divergence free velocity $\nabla\cdot\mathbf{u}=\mathbf{0}$. Therefore, an identity for the second-order tensor field $\boldsymbol{L}$ and a solenoidal vector field $\mathbf{v}$ can be derived by employing the tensor calculus and Einstein's summation convention (3.10).

$$\begin{aligned}
\nabla\cdot(\boldsymbol{L}\otimes\mathbf{v}) &= \frac{\partial}{\partial x_l}\left((\boldsymbol{L}_{ij}\boldsymbol{e}_i\otimes\boldsymbol{e}_j)\otimes(v_k\boldsymbol{e}_k)\right)\cdot\boldsymbol{e}_l \\
&= \left(L_{ij}v_k\right)_{,l}\boldsymbol{e}_i\otimes\boldsymbol{e}_j\delta_{kl} \\
&= \left(L_{ij,k}v_k + L_{ij}v_{k,k}\right)\boldsymbol{e}_i\otimes\boldsymbol{e}_j \\
&= \mathbf{v}\cdot\nabla\boldsymbol{L} + (\nabla\cdot\mathbf{v})\,\boldsymbol{L}
\end{aligned} \tag{3.10}$$

In equation (3.10), the dyadic product of two vectors is represented by $\otimes$, $\boldsymbol{e}_l$ for $l\in\{1,...,d\}$ denotes the (cartesian) basis vectors of $\mathbb{R}^d$ and derivative w.r.t the corresponding spatial dimension is indicated by the subscript $(,l)$. By setting $\boldsymbol{L}=\mathbf{D}(\mathbf{u})$ and $\mathbf{v}=\mathbf{u}$ in equation (3.3) leads to:

$$\overset{\nabla}{\mathbf{D}} = \frac{\partial}{\partial t}\mathbf{D}(\mathbf{u}) + \nabla\cdot(\mathbf{D}(\mathbf{u})\otimes\mathbf{u}) - \nabla\mathbf{u}^T\cdot\mathbf{D}(\mathbf{u}) - \mathbf{D}(\mathbf{u})\cdot\nabla\mathbf{u} \tag{3.11}$$

This formulation offers the possibility to shift the higher order derivatives onto the test function in the weak formulation via integration by parts.

3.2 Tensor Stokes formulation

The novel Tensor Stokes formulation introduced by Westervoss [66] tends to aid the numerical simulations of the viscoelastic fluids with special emphasis on the non-solvent case. The underlying assumption of this approach is the existence of a decomposition of the extra-stress tensor

$$\boldsymbol{S} = \boldsymbol{T} \cdot \mathbf{D}(\mathbf{u}) \tag{3.12}$$

where $\boldsymbol{T} = \boldsymbol{T}(\mathbf{u}, \nabla\mathbf{u})$ is the second-order tensor called the Diffusion Tensor and $\mathbf{D}(\mathbf{u})$ is the deformation tensor. The idea suggests that the complex hidden rheology of the viscoelastic model can be decomposed into the product of deformation tensor and the implicitly defined tensor-valued viscosity function. The addition of such decomposition to the basic viscoelastic model results in a number of variants of the Tensor Stokes formulations each with it's own potential advantages.

The two-field Stokes flow offers the possibility to remove the complex rheology of the fluid from the system of equations and establish a straightforward numerical treatment of the viscoelastic fluids. For simple flows, this approach can reduce the three field viscoelastic model to a generalized "Tensor Stokes" problem, thus eliminating the requirement for a distinct stress tensor as a third field. The two field Tensor Stokes flow is indeed one of the most exciting case as it is expected that this approach can reproduce the viscoelastic flow characteristics that are typically obtained using the three field formulation. Most of the problems that the scientific community faced in the numerical treatment of viscoelastic flows (described above) have already been solved by the development of such formulation. Introducing the stress decomposition (3.12) into equation (2.3) results in the modified Stokes problem or so called the Tensor Stokes formulation given as follows:

$$\begin{aligned} -\nabla \cdot (\boldsymbol{T}(\mathbf{u}) \cdot \mathbf{D}(\mathbf{u})) + \nabla p &= 0 \\ \nabla \cdot \mathbf{u} &= 0 \end{aligned} \tag{3.13}$$

The idea is to consider the standard viscoelastic formulation (2.4) and compute the differential solution $(\mathbf{u}_{\text{diff}}, p_{\text{diff}}, \boldsymbol{S}_{\text{diff}})$. The next step is to determine the tensor $\boldsymbol{T}$. This can be done by utilizing the splitting $\boldsymbol{S}_{\text{diff}} = \boldsymbol{T} \cdot \mathbf{D}(\mathbf{u}_{\text{diff}})$ or one could prescribe an analytic $\boldsymbol{T}$ (if known) or externally calculated flow profiles of $\boldsymbol{T}$ can also be applied. Solve the problem (3.13) with the given tensor $\boldsymbol{T}$ for $(\mathbf{u}_{\text{TSt}}, p_{\text{TSt}})$. It is expected that the result obtained from the differential viscoelastic formulation is similar to the Tensor Stokes formulation i.e. $\mathbf{u}_{\text{diff}} \sim \mathbf{u}_{\text{TSt}}$ and $p_{\text{diff}} \sim p_{\text{TSt}}$. The Tensor Stokes formulation (3.13) leads to the introduction of the modified diffusive operator in the momentum equation. The momentum equation yields the tensor $\boldsymbol{T}$, which is a non-symmetric tensor-valued viscosity. Furthermore, as already discussed in *Chapter 2*, the introduction of the symmetrized version of the Tensor Stokes formulation (3.14) led to the improved numerical behavior also for the two-field formulation.

$$
\begin{aligned}
-\frac{1}{2}\nabla \cdot \left(\boldsymbol{T} \cdot \mathbf{D}\left(\mathbf{u}\right) + \mathbf{D}\left(\mathbf{u}\right) \cdot \boldsymbol{T}^{T}\right) + \nabla p &= 0 \\
\nabla \cdot \mathbf{u} &= 0
\end{aligned}
\tag{3.14}
$$

The diffusion tensor or the tensorial viscosity in equation (3.14) could be in two $\boldsymbol{T} \in \mathbb{R}^{2\times 2}$ or in three dimensions $\boldsymbol{T} \in \mathbb{R}^{3\times 3}$. The viscoelastic flows where the solvent viscosity is present i.e., $\eta_s \neq 0$ a term $2\eta_s \boldsymbol{I}$ is added to the Tensor Stokes operator in the momentum equation as $\tilde{\boldsymbol{T}} = 2\eta_s \boldsymbol{I} + \boldsymbol{T}$ rather than $\boldsymbol{T}$. The Tensor Stokes formulation has many advantages, and its numerical application weakens or perhaps eliminates the significant difficulties brought on by the numerical simulations of viscoelastic flows. In a perfect scenario, the tensorial viscosity $\boldsymbol{T}$ would be known or provided. In accordance with the differential viscoelastic model the analytic expression for $\boldsymbol{T}$ depends on the velocity or its gradient. The Tensor Diffusion formulation (3.14) is obtained in the primitive variables pressure and velocity on the availability of the tensor $\boldsymbol{T}$. It is clear from problem (3.14) that the diffusive operator, which affects velocity, is integrated back into the momentum equation even in the situation of zero solvent viscosity. As the rheological effects are incorporated into the momentum equation via tensor $\boldsymbol{T}$ the primary benefit of this formulation is the computation of viscoelastic solutions using a two-field formulation instead of three-field formulation.

The extra-stress tensor $\boldsymbol{S}$ can be calculated by the post-processing of the results $\mathbf{u}, p$ obtained from (3.14). The findings of Westervoss [66] aided and eased the numerous constraints faced by the scientific community in the numerical simulations of the viscoelastic flows. The additional stability requirement between the velocity and the stress does not need to be taken into consideration when using the Tensor-Stokes formulation (3.14) because there is no need to solve for a stress variable.

In the perfect scenario, one could define the complex hidden rheology of the viscoelastic fluid in terms of the tensorial viscosity $\boldsymbol{T}$, which was previously represented by the constitutive equation, the usual three field formulation is reduced to the two field formulation in $\mathbf{u}, p$. Additionally, the modified Stokes problem (3.14) can be treated by using accurate, robust and efficient numerical techniques.

Moreover, the modified Stokes flow would be suitable to the same solution approaches described for the Stokes flow. In doing so, the computational work necessary to solve the three field viscoelastic formulation would be greatly reduced. Also, by doing this, a significant problem brought on by the constitutive equation's convective term needs not to be taken into account. That means the two field Tensor Stokes formulation has a plethora of unique advantages. However, it's not that simple to have a known tensorial viscosity for highly complex rheological models and unfortunately, the explicit modeling of the tensorial viscosity $\boldsymbol{T}$ for general or complex flow configurations has not been accomplished up until this point. As a result, for the complicated fluids, the original differential viscoelastic model (2.4) must be used to calculate the tensorial viscosity $\boldsymbol{T}$ numerically.

Using the steady state standard three field viscoelastic extra stress tensor formulation for the zero solvent viscosity, one might calculate the tensorial viscosity numerically in the first step. In the current formulation, an additional equation in $\boldsymbol{T}$ is considered. The steady state set of equations reads as follows:

$$-\nabla \cdot \boldsymbol{S} + \nabla p = 0 \tag{3.15a}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{3.15b}$$

$$\lambda \overset{\nabla}{\boldsymbol{S}} + f(\boldsymbol{S}, \lambda)\, \boldsymbol{S} + h(\boldsymbol{S}) = 2\eta_p \mathbf{D}(\mathbf{u}) \tag{3.15c}$$

$$\boldsymbol{S} = \boldsymbol{T} \cdot \mathbf{D}(\mathbf{u}) \tag{3.15d}$$

The set of equations in (3.15) is obtained by the addition of an algebraic equation to (2.4). The tensor based viscosity $\boldsymbol{T}$ is computed by the post processing of the $\mathbf{u}, p$ and $\boldsymbol{S}$.
Another version of the Tensor Stokes formulation introduced by Westervoss [66] is the actual introduction of the symmetric version of the tensor based viscosity $\boldsymbol{T}$ in the momentum equation (3.15a). The steady state set of equations reads as follows:

$$-\frac{1}{2}\nabla \cdot \left(\boldsymbol{T} \cdot \mathbf{D}(\mathbf{u}) + \mathbf{D}(\mathbf{u}) \cdot \boldsymbol{T}^T\right) + \nabla p = 0 \tag{3.16a}$$

$$\nabla \cdot \mathbf{u} = 0 \tag{3.16b}$$

$$\lambda \overset{\nabla}{\boldsymbol{S}} + f(\boldsymbol{S}, \lambda)\, \boldsymbol{S} + h(\boldsymbol{S}) = 2\eta_p \mathbf{D}(\mathbf{u}) \tag{3.16c}$$

$$\boldsymbol{S} = \boldsymbol{T} \cdot \mathbf{D}(\mathbf{u}) \tag{3.16d}$$

The two formulations (3.15) and (3.16) are equivalent on the continuous level but not on the discrete level. Due to the presence of the Tensor Stokes operator in the momentum equation, the velocity, stress, and pressure solutions are coupled with the tensorial viscosity $\boldsymbol{T}$ in set of equations (3.16). By doing this, it can be seen that the formulation's problem size has grown to four equations. Detailed research has been done on the four field formulation by Westervoss [66] and their analysis of the formulation helped to form a new version of the Tensor Stokes formulation.
In order to compute the tensor-based viscosity in terms of the three field viscoelastic differential model, the next variant of the Tensor Stokes formulation presents an enhanced and sophisticated numerical algorithm. Westervoss [66] proposed an advanced method to compute the tensorial viscosity while looking into the hidden rheology within the tensorial viscosity. The goal is to use the stress decomposition not only in the momentum equation but also in the constitutive equation of the three field formulation. Therefore, in problem (2.4) the stress tensor is replaced everywhere by its decomposition. As a result, one obtain the three field Tensor Stokes formulation in $\mathbf{u}, p, \boldsymbol{T}$. The novel three field Tensor Stokes approach is given as,

$$\rho\frac{\partial \mathbf{u}}{\partial t} - \frac{1}{2}\nabla\cdot\left(\boldsymbol{T}\cdot\mathbf{D}\left(\mathbf{u}\right)+\mathbf{D}\left(\mathbf{u}\right)\cdot\boldsymbol{T}^{T}\right)+\nabla p = 0 \tag{3.17a}$$

$$\nabla\cdot\mathbf{u} = 0 \tag{3.17b}$$

$$\lambda(\overset{\nabla}{\boldsymbol{T}}\cdot\mathbf{D}) + f(\boldsymbol{T}\cdot\mathbf{D},\lambda)\left(\boldsymbol{T}\cdot\mathbf{D}\right) + h\left(\boldsymbol{T}\cdot\mathbf{D}\right) = 2\eta_p\mathbf{D}\left(\mathbf{u}\right) \tag{3.17c}$$

where

$$\begin{aligned}(\overset{\nabla}{\boldsymbol{T}}\cdot\mathbf{D}) = \frac{\partial}{\partial t}\left(\boldsymbol{T}\cdot\mathbf{D}\left(\mathbf{u}\right)\right) + \left(\mathbf{u}\cdot\nabla\right)\left(\boldsymbol{T}\cdot\mathbf{D}\left(\mathbf{u}\right)\right) - \\ \left(\nabla\mathbf{u}^{T}\cdot\left(\boldsymbol{T}\cdot\mathbf{D}\left(\mathbf{u}\right)\right) + \left(\boldsymbol{T}\cdot\mathbf{D}\left(\mathbf{u}\right)\right)\cdot\nabla\mathbf{u}\right)\end{aligned} \tag{3.18}$$

The tensorial viscosity is represented by its constitutive equation in the viscoelastic formulation (3.17). Given that the tensor-based viscosity is calculated as a constitutive equation that is a crucial component of the three field viscoelastic formulation, it is anticipated that the this technique will perform better than its version mentioned in (3.16). Moreover, the reduced size of the formulation is a definite benefit of the later over former in terms of computational cost. Hence, the constitutive equation for the tensor based viscosity seems to be quite attractive and will be studied in detail. Also, from an practical standpoint, the formulation (3.17) looks to be rather appealing for handling the different viscoelastic fluids of relevance. However, similar to the EVSS formulation, one may observe the appearance of the second order velocity derivatives in the convective component of equation (3.18). One approach could be to treat these derivatives numerically and add the deformation tensor as an additional variable. The aim is to prevent this because it would expand the problem size to four field formulation. The similar approach (3.10) is employed to treat the second order velocity derivative in the Tensor Stokes formulation. By setting $\boldsymbol{L} = \boldsymbol{T}\cdot\mathbf{D}\left(\mathbf{u}\right)$ and $\mathbf{v} = \mathbf{u}$ in (3.10) the equation (3.18) reads as,

$$\begin{aligned}(\overset{\nabla}{\boldsymbol{T}}\cdot\mathbf{D}) = \frac{\partial}{\partial t}\left(\boldsymbol{T}\cdot\mathbf{D}\left(\mathbf{u}\right)\right) + \nabla\cdot\left[\left(\boldsymbol{T}\cdot\mathbf{D}\left(\mathbf{u}\right)\right)\otimes\mathbf{u}\right] - \\ \left(\nabla\mathbf{u}^{T}\cdot\left(\boldsymbol{T}\cdot\mathbf{D}\left(\mathbf{u}\right)\right) + \left(\boldsymbol{T}\cdot\mathbf{D}\left(\mathbf{u}\right)\right)\cdot\nabla\mathbf{u}\right)\end{aligned} \tag{3.19}$$

Similar to the EVSS formulation, (3.19) offers the possibility to shift the higher order derivatives onto the test function in the weak formulation via integration by parts.

3.3 Summary

As discussed earlier, the aim of this research work is to obtain the solution of the three-field viscoelastic models with pure polymer melts i.e. zero solvent viscosity. Considering the standard viscoelastic formulation for such purpose, it is impossible to obtain the solution in a decoupled way without any added stabilization because of the absence of the velocity coupling in the momentum equation. Also, the monolithic approach for such formulation

is quite challenging, especially for the multigrid solver. Therefore, the stress splitting is introduced with the aim to recover the velocity coupling back into the momentum equation. The change of variables is performed in the standard viscoelastic model to obtain the EVSS and Tensor Stokes formulation. The next issue is the appearance of the second order velocity derivatives in the convective term of the constitutive equation of both the formulations. In literature, the usual way to treat this issue is to consider the deformation tensor as an additional field thus increasing the problem size to four field formulation and therefore, increasing the numerical cost.

In this thesis, we have reformulated the convective term by explicitly considering the divergence-free nature of the velocity field. The proposed identity based on the tensor calculus helped to retain the problem size to three field formulation with the ability to move the higher order derivatives to the test function in the weak formulation. The two formulations under consideration, EVSS and Tensor Stokes formulation, have the potential advantage of being able to be solved for the viscoelastic fluids with zero solvent viscosity due to the recovery of the velocity coupling back into the momentum equation. Furthermore, these formulations offer the possibility to be treated in a decoupled and fully monolithic manner.

4

Finite element method

Computational Fluid Dynamics (CFD) uses a set of algebraic equations to replace the mathematical model represented by partial differential equations (PDEs). These algebraic equations are subsequently solved using modern computers and a reliable numerical approach. A very flexible numerical technique used to determine an approximate solution of the physical problem is the finite element method (FEM). All the numerical results in this thesis, have been obtained using the FEATFLOW (Finite Element Analysis Tool for flow problems) software.
The finite element method is numerical method for finding the approximate solutions to the partial differential equations. FEM is a well-known, effective method that is known for its adaptability in handling any complex PDEs and geometry. FEM, as opposed to FDM, can handle extremely complicated geometries, making it an excellent choice for performing numerical simulations. The numerical solution of the viscoelastic formulations (described in *chapter 2*) with finite element discretization is based on its variational or equivalently called weak formulation. For this purpose, the following sub-spaces of the usual Lebesgue function space $L^2(\Omega)$ of square integrable functions are defined as follows:

$$\begin{aligned} L_0^2(\Omega) &= \{q : q \in L^2(\Omega) \; with \int_\Omega q(x) d\Omega = 0, x \in \Omega\} \\ H^1(\Omega) &= \{\boldsymbol{v} : \boldsymbol{v} \in L^2(\Omega) \; , \; \partial_i \boldsymbol{v} \in L^2(\Omega) \; , \; 1 \leq i \leq d\} \\ H_0^1(\Omega) &= \{\boldsymbol{v} : \boldsymbol{v} \in L^2(\Omega) \; , \; \partial_i \boldsymbol{v} \in L^2(\Omega) \; with \; \boldsymbol{v} = 0 \; on \; \Gamma\} \end{aligned} \tag{4.1}$$

The spaces defined above are the Hilbert spaces. The unknown field variables for the velocity, pressure and stress in the described formulations are approximated in vector-scalar-tensor valued function spaces $\boldsymbol{V} \times \boldsymbol{Q} \times \boldsymbol{X}$, respectively. The particular choice of the function spaces $\boldsymbol{V}$ and $\boldsymbol{Q}$ is well-established, when considering the weak formulation of the standard Stokes equations. This formulation arises by ignoring the viscoelastic effects and subsequently the constitutive equation. In this regard, $\boldsymbol{V} := (H_0^1(\Omega))^2$ and $\boldsymbol{Q} := L^2(\Omega)$ which is the usual Sobolev and Lebesgue spaces are selected for the velocity

and pressure fields, respectively [42, 54, 58, 10]. Alternatively, the Stokes flow can be expressed in terms of three field formulation by setting $\lambda = 0, \eta_s = 0$ and $\eta_p > 0$. Here, $\boldsymbol{V}$ and $\boldsymbol{Q}$ are selected in the similar way, and the usual choice for the stress tensor is $\boldsymbol{X} := (L^2(\Omega))^{2\times 2}$ [42, 7, 55, 52]. While considering the viscoelastic effects that means $\lambda > 0$, the stress tensor needs to be chosen from the subset $\boldsymbol{X} \subset (L^2(\Omega))^{2\times 2}$ because of the presence of the convective term in the constitutive equation.

4.1 Weak formulation

The first step in FEM is to convert the set of PDEs representing the physical system to the variational or the weak formulation, that transfers the derivatives to the test functions. This method relaxes the continuity restrictions by testing the strong form with an arbitrary function known as the test function. If the test functions $(\boldsymbol{v}, q, \boldsymbol{\phi})$ belong to the same function space $\boldsymbol{V} \times \boldsymbol{Q} \times \boldsymbol{X}$ then the formulation is called the Galerkin formulation [20]. Following are the steps required to obtain the weak formulation from the strong formulation [8],

- Multiplication of the momentum equation by the test function $\boldsymbol{v} \in \boldsymbol{V}$, continuity equation by $q \in \boldsymbol{Q}$ and stress equation by $\boldsymbol{\phi} \in \boldsymbol{X}$.
- Integrate over the domain and apply integration by parts.

The variational formulation of extra-stress tensor, EVSS and the Tensor Stokes formulation are given below.

Extra-stress tensor weak formulation

Consider the system of equations (2.4) and introduce three test functions ($\boldsymbol{v}, q$ and $\boldsymbol{\phi}$). The equations in the weak formulation after applying integration by parts reads as follows:

$$\begin{aligned} \rho \int_\Omega \frac{\partial \mathbf{u}}{\partial t} \boldsymbol{v}\, dx + 2\eta_s \int_\Omega \mathbf{D}(\mathbf{u}) : \mathbf{D}(\boldsymbol{v})\, dx - \int_\Omega p \nabla \cdot \boldsymbol{v}\, dx + \int_\Omega \boldsymbol{S} : \mathbf{D}(\boldsymbol{v})\, dx = 0 \\ \int_\Omega q\, \nabla \cdot \mathbf{u}\, dx = 0 \\ \int_\Omega \left(\lambda \overset{\nabla}{\boldsymbol{S}} + f(\boldsymbol{S}, \lambda)\, \boldsymbol{S} + h(\boldsymbol{S}) \right) \boldsymbol{\phi}\, dx - \int_\Omega 2\eta_p \mathbf{D}(\mathbf{u}) : \boldsymbol{\phi}\, dx = 0 \end{aligned} \tag{4.2}$$

EVSS weak formulation

Consider the system of equations (2.12) along with the identity (3.10) employed in equation (3.11) and introduce three test functions ($\boldsymbol{v}, q$ and $\boldsymbol{\phi}$). The weak form of the EVSS formulation reads as follows:

$$\begin{aligned}
\rho \int_\Omega \frac{\partial \mathbf{u}}{\partial t} \boldsymbol{v}\, dx + 2\eta_0 \int_\Omega \mathbf{D}(\mathbf{u}) : \mathbf{D}(\boldsymbol{v})\, dx - \int_\Omega p \nabla \cdot \boldsymbol{v}\, dx + \int_\Omega \boldsymbol{E} : \mathbf{D}(\boldsymbol{v})\, dx = 0 \\
\int_\Omega q\, \nabla \cdot \mathbf{u}\, dx = 0 \\
\int_\Omega f\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}, \lambda\right)\left(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}\right) : \boldsymbol{\phi}\, dx + \\
\int_\Omega \lambda (2\eta_p \overset{\nabla}{\mathbf{D}}(\mathbf{u}) + \overset{\nabla}{\boldsymbol{E}}) : \boldsymbol{\phi}\, dx + \int_\Omega h\left(2\eta_p \mathbf{D}(\mathbf{u}) + \boldsymbol{E}\right) : \boldsymbol{\phi}\, dx - \int_\Omega 2\eta_p \mathbf{D}(\mathbf{u}) : \boldsymbol{\phi}\, dx = 0
\end{aligned}$$

where,

$$\begin{aligned}
\int_\Omega \overset{\nabla}{\mathbf{D}}(\mathbf{u}) : \boldsymbol{\phi}\, dx = \int_\Omega \frac{\partial \mathbf{D}(\mathbf{u})}{\partial t} : \boldsymbol{\phi}\, dx - \int_\Omega (\mathbf{D}(\mathbf{u}) \otimes \mathbf{u}) : \nabla \boldsymbol{\phi}\, dx \\
- \int_\Omega \left(\nabla \mathbf{u}^T \cdot \mathbf{D}(\mathbf{u}) + \mathbf{D}(\mathbf{u}) \cdot \nabla \mathbf{u}\right) : \boldsymbol{\phi}\, dx
\end{aligned} \tag{4.3}$$

Tensor Stokes weak formulation

Consider the PDEs in (2.22) along with the identity (3.10) employed in equation (3.19). Introduce and multiply the test functions ($\boldsymbol{v}, q$ and $\boldsymbol{\phi}$) leads to the following weak formulation of the Tensor Stokes problem.

$$\begin{aligned}
\rho \int_\Omega \frac{\partial \mathbf{u}}{\partial t} \boldsymbol{v}\, dx + \frac{1}{2} \int_\Omega \left(\boldsymbol{T} \cdot \mathbf{D}(\mathbf{u}) + \mathbf{D}(\mathbf{u}) \cdot \boldsymbol{T}^T\right) : \mathbf{D}(\boldsymbol{v})\, dx - \int_\Omega p \nabla \cdot \boldsymbol{v}\, dx = 0 \\
\int_\Omega q\, \nabla \cdot \mathbf{u}\, dx = 0 \\
\lambda \int_\Omega (\boldsymbol{T} \cdot \overset{\nabla}{\mathbf{D}}(\mathbf{u})) : \boldsymbol{\phi}\, dx + \int_\Omega f(\boldsymbol{T} \cdot \mathbf{D}, \lambda)(\boldsymbol{T} \cdot \mathbf{D}) : \boldsymbol{\phi}\, dx + \int_\Omega h(\boldsymbol{T} \cdot \mathbf{D}) : \boldsymbol{\phi}\, dx = \\
\int_\Omega 2\eta_p \mathbf{D}(\mathbf{u}) \ : \boldsymbol{\phi}\, dx
\end{aligned} \tag{4.4}$$

where,

$$\begin{aligned}
\int_\Omega (\boldsymbol{T} \cdot \overset{\nabla}{\mathbf{D}}(\mathbf{u})) : \boldsymbol{\phi}\, dx = \int_\Omega \frac{\partial}{\partial t}(\boldsymbol{T} \cdot \mathbf{D}) : \boldsymbol{\phi}\, dx - \int_\Omega ((\boldsymbol{T} \cdot \mathbf{D}) \otimes \mathbf{u}) : \nabla \boldsymbol{\phi}\, dx \\
- \int_\Omega \left(\nabla \mathbf{u}^T \cdot (\boldsymbol{T} \cdot \mathbf{D}(\mathbf{u})) + (\boldsymbol{T} \cdot \mathbf{D}(\mathbf{u})) \cdot \nabla \mathbf{u}\right) : \boldsymbol{\phi}\, dx
\end{aligned} \tag{4.5}$$

4.2 Finite element discretization

The next step in finite element method is to replace the infinite-dimensional spaces $\boldsymbol{V}, \boldsymbol{Q}$ and $\boldsymbol{X}$ by the finite-dimensional spaces $\boldsymbol{V}_h, \boldsymbol{Q}_h$ and $\boldsymbol{X}_h$. If $\boldsymbol{V}_h \subset \boldsymbol{V}$, $\boldsymbol{Q}_h \subset \boldsymbol{Q}$ and $\boldsymbol{X}_h \subset \boldsymbol{X}$ the finite element method is called conforming otherwise it is called non-conforming. In this work only the conforming finite element method is considered. The finite element expansion of the discrete solutions for $\mathbf{u}_h, p_h, \boldsymbol{S}_h$ ($\boldsymbol{E}_h$ in case of EVSS and $\boldsymbol{T}_h$ in case of Tensor Stokes) are given as,

$$\mathbf{u}_h = \sum_{i=1}^{N_u} u_i \boldsymbol{v}_i, \quad p_h = \sum_{j=1}^{N_p} p_j q_j, \quad \boldsymbol{S}_h = \sum_{k=1}^{N_S} S_k \boldsymbol{\phi}_k \tag{4.6}$$

where N is number of degrees of freedom on each mesh level for each variable. In equation (4.6), $\boldsymbol{v}_i, q_j$ and $\boldsymbol{\phi}_k$ are the vector, scalar and tensor valued basis functions, respectively. Moreover, $\mathbf{u} = (u_i)_{i=1}^{N_u}$, $p = (p_j)_{j=1}^{N_p}$ and $\boldsymbol{S} = (S_k)_{k=1}^{N_S}$ represents the vectors that contain the degrees of the freedom of the field variables in the discretized domain.
The next step involves selecting the appropriate basis (test) functions for our formulation. In the viscoelastic formulation for pure polymer melts, it is necessary to satisfy compatibility conditions not only between velocity and pressure but also between velocity and stress. Firstly, we would discuss the finite element pair for the Stokes problem and later on for the three field mixed formulation. While selecting the approximating spaces for the velocity and pressure, following conditions are to be considered,

1. Number of degrees of freedom for velocity are greater than number of pressure degrees of freedom, only necessary
2. The spaces $\boldsymbol{V}_h$ and $\boldsymbol{Q}_h$ should satisfy the discrete inf-sup condition or LBB condition [27] .

The well-known LBB *(Ladyzhenskaya-Babuska-Brezzi)* condition or inf-sup condition between the velocity and pressure space must be satisfied in order to select a stable finite element pair for the Navier Stokes or Stokes problem. The LBB condition in the discrete form is given as,

$$\sup_{\mathbf{v} \in \mathbf{v}_h} \frac{\int_\Omega \operatorname{div} \mathbf{v}\, q\, dx}{||\mathbf{v}||_{1,\Omega}} \geq \epsilon ||q||_{0,\Omega} \quad \forall\, q \ \in \ \boldsymbol{Q}_h \tag{4.7}$$

where $\epsilon > 0$ is the mesh independent constant and $||.||_{0,\Omega}$ is the norm in Lebesgue space $L^2(\Omega)$. There are a vast number of finite element spaces satisfying the inf-sup condition for the velocity pressure coupling. The well-known Q_2/P_1^{disc} satisfies the LBB condition for the velocity-pressure formulation and is known to be one of the most popular Stokes element since it is a quite accurate and robust finite element pair for highly viscous incompressible

flow [3, 30]. The choice of approximation spaces for the velocity and stress is also subject to additional compatibility condition when working with the mixed (velocity-pressure-stress) formulation.

$$\sup_{\boldsymbol{S}\in\mathbf{X}_h} \frac{\int_\Omega \text{div } \boldsymbol{S}\ \mathbf{v} dx}{||\boldsymbol{S}||_{0,\Omega}} \geq \beta||\mathbf{v}||_{1,\Omega} \quad \forall\ \mathbf{v}\ \in\ \boldsymbol{V}_h \tag{4.8}$$

where β is the mesh-independent constant, $||.||_{1,\Omega}$ is the norm in Sobolev space $H^1(\Omega)$. Finite element pairs are considered to be compatible if they meet the compatibility conditions in equation (4.7) and (4.8). Selection of a suitable pair for the three field viscoelastic formulation is a challenging task. Fortin and Pierre [24] have shown that in the absence of the viscous contribution, the standard discrete spaces have to satisfy the following conditions:

1. The velocity-pressure spaces must be compatible with respect to equation (4.7)
2. If the stress tensor $\boldsymbol{S}$ is approximated by discontinuous FEM, the deformation tensor must be a member of the same space

$$\mathbf{D}(\mathbf{u}_h) = \frac{1}{2}(\nabla\mathbf{u}_h + \nabla\mathbf{u}_h^T)\ \in\ \boldsymbol{X}_h \qquad \forall\ \mathbf{u} \in\ \boldsymbol{V}_h \tag{4.9}$$

3. If the stress tensor is approximated by continuous FEM, the number of local degrees of freedom must be larger than that used for the velocity space.

The first condition is already discussed. Regarding the second and third condition the choice is influenced whether the stress is approximated in continuous or discontinuous finite element space. The discontinuous Galerkin technique used by Fortin and Fortin [22] satisfies the LBB requirement, if the stress tensor is approximated in the discontinuous space as shown in Fig. 4.1.

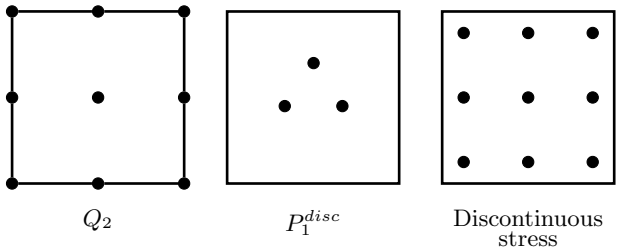


Fig. 4.1: Bi-quadratic velocity; discontinuous linear pressure element; discontinuous quadratic stress element of Fortin and Fortin (From Left to Right).

The selection of the finite element pair meeting the third requirement have been the subject of substantial research by Marchal and Crochet [42]. They approximated the velocity

and pressure solutions by employing the biquadratic and bilinear elements, respectively. For the stress tensor, their findings supported the fact that the number of stress degrees of freedom should be greater than the number of velocity degrees of freedom, if the stress tensor is approximated in the continuous finite element space. To increase the local degrees of freedom for the stress space, they proposed the subcell discretization. They approximated the stress via $n \times n$ bilinear subelements of the approximation of the velocity field, where n=2, 3 or 4 as shown in Fig. 4.2. They obtained the stable numerical results for n=3 and 4 on the four to one abrupt contraction and the stick-slip problem.

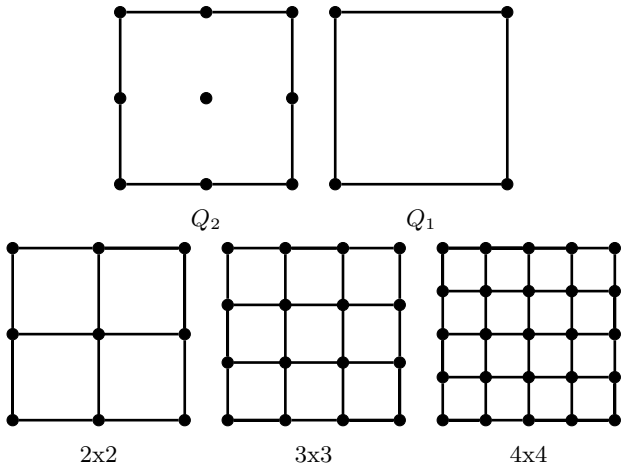


Fig. 4.2: Bi-quadratic velocity and bi-linear pressure element pair (Top); bi-linear stress sub-elements (Bottom) of Marchal and Crochet [42].

A detailed review of mixed finite element method can be found in the work of Baaijens [5, 6]. In the current research work, the stress tensor is approximated using the continuous finite element method (FEM). Consequently, the finite element pair must be chosen that satisfy the LBB condition. The selected choice for the velocity, pressure and stress finite element are as follows;

- The velocity space belongs to the nine-node Lagrangian elements with biquadratic shape functions i.e., Q_2 finite element.
- P_1^{disc} finite element is selected for the pressure.
- The stress space belongs to the sixteen-node Lagrangian elements with bicubic shape functions i.e., Q_3 finite element.

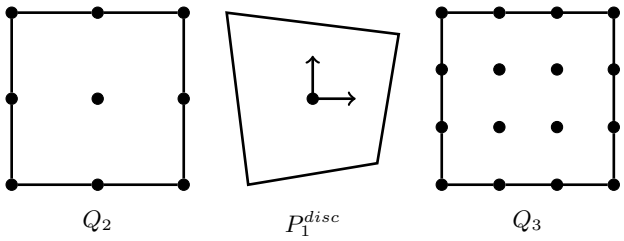


Fig. 4.3: Finite elements Q_2,P_1^{disc},Q_3 for velocity, pressure and stress, respectively.

The flow domain is discretized by means of quadrilaterals. It can be seen in Fig. 4.3 that there are nine local degrees of freedom in the Q_2 finite element space, four of which are located at the vertices, four at the mid point of each edge, and one at the quadrilateral's center. There are 16 local degrees of freedom in Q_3, four of which are located at the vertices and two are located at each edge in an equidistant position from each other, that counts total to 12 degrees of freedom. Four are located at the quadrilateral's center and are equidistant from each other, as are the degrees of freedom located at the edges. P_1^{disc} has three local degrees of freedom located at the center of the quadrilateral with the linear function values and its partial derivatives.

A triangulation $\mathcal{T}_h$ of the two-dimensional computational domain Ω is defined, comprising discrete quadrilateral elements $T_1, T_2, ...T_M$. A mapping $\phi_T : T_{ref} \rightarrow T$ between the real $T \in T_h$ and reference element $T_{ref} = [-1, 1]^2$ is taken into account when defining the polynomial spaces. The polynomial spaces on the quadrilateral T are defined as,

$$\begin{aligned} Q_2(T) &= \{q \; o \; \phi_T^{-1} \mid q \in span < 1, x, y, xy, x^2, y^2, x^2y, xy^2, y^2x^2 >\} \\ Q_3(T) &= \{q \; o \; \phi_T^{-1} \mid q \in span < 1, x, y, xy, x^2, y^2, x^2y, xy^2, y^2x^2, x^3, \\ &\qquad y^3, xy^3, x^2y^3, x^3y, x^3y^2, x^3y^3 >\} \\ P_1(T) &= \{q \; o \; \phi_T^{-1} \mid q \in span < 1, x, y >\} \end{aligned} \tag{4.10}$$

The velocity, pressure and stress solutions are approximated in the following discrete spaces:

$$\begin{aligned} \boldsymbol{V}_h &= \{\boldsymbol{v}_h \in \boldsymbol{V}, \boldsymbol{v}_{h|T} \in (Q_2(T))^2\} \\ Q_h &= \{q_h \in Q, q_{h|T} \in \left(P_1^{disc}(T)\right)\} \\ \boldsymbol{X}_h &= \{\boldsymbol{\phi}_h \in \boldsymbol{X}, \boldsymbol{\phi}_{h|T} \in (Q_3(T))^{2\times 2}\} \end{aligned} \tag{4.11}$$

The use of the aforementioned discrete spaces ensures a stable approximation of the compatibility condition between the velocity and pressure, satisfying the LBB condition. Additionally, by employing the higher-order Q_3 finite element approximation for the stress, the compatibility condition between velocity and stress is achieved for the pure polymer melts i.e., $\eta_s = 0$.

For $\eta_s > 0$, the need to satisfy the compatibility condition between velocity and stress

vanishes. However, we will stick to Q_3 finite element for stress due to its high accuracy. For the pure polymer melts, we are successful in achieving a stable finite element triplet $Q_2/P_1^{disc}/Q_3$ for our formulation however sometimes for the complex viscoelastic flows or the convection-dominated flows an additional stabilization may be necessary to avoid any convergence issues. That means as the viscoelastic effects increases one may need to add stabilization to the problem to aid the convergence. In this regard, Edge Oriented stabilization may be considered [61].

4.3 Edge Oriented stabilization

In the Edge Oriented (EO) stabilization, a jump term of the solution gradient across the element edges is added either to the velocity or stress tensor equations. This term controls the solution jumps between the edges of the adjacent elements [13]. The jump term [45, 61, 62] for the velocity and stress can be defined as follows:

$$J_{\mathbf{u}} = \sum_{E} \gamma_{\mathbf{u}} h^p \int_E [\mathbf{D}(\mathbf{u})] : [\mathbf{D}(\boldsymbol{v})] ds \tag{4.12}$$

$$J_{\boldsymbol{S}} = \sum_{E} \gamma_{\boldsymbol{S}} h^p \int_E [\nabla \boldsymbol{S}] : [\nabla \boldsymbol{\phi}] ds \tag{4.13}$$

where $\gamma_{\mathbf{u}}$ and $\gamma_{\boldsymbol{S}}$ are the mesh independent scaling factor for the jump stabilization operator, h is the mesh refinement size and the exponent p is the power of approximation order [61]. The notation in the form $[\boldsymbol{\epsilon}]$ defines the jumps over an edge E of the element i.e. it computes the difference of the gradient values $[\boldsymbol{\epsilon}] = \boldsymbol{\epsilon}^+ - \boldsymbol{\epsilon}^-$. By the addition of the EO stabilization to the discrete system the structure of the matrix remains the same however the sparsity pattern of the matrix does not remain the same. The discrete system of equations for the three field EVSS and Tensor Stokes formulation along with the addition of EO stabilization term in the velocity and stress are given as follows:

$$\begin{bmatrix} A + J_{\mathbf{u}} & B & R \\ B^T & 0 & 0 \\ K_{\nabla_{\mathbf{D}(\mathbf{u})}} & 0 & Z + J_{\boldsymbol{E}} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \\ \boldsymbol{E} \end{bmatrix} = \begin{bmatrix} \text{rhs}_{\mathbf{u}} \\ \text{rhs}_p \\ \text{rhs}_{\boldsymbol{E}} \end{bmatrix} \tag{4.14}$$

$$\begin{bmatrix} A_\mu + J_{\mathbf{u}} & B & 0 \\ B^T & 0 & 0 \\ K & 0 & M + J_{\boldsymbol{T}} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \\ \boldsymbol{T} \end{bmatrix} = \begin{bmatrix} \text{rhs}_{\mathbf{u}} \\ \text{rhs}_p \\ \text{rhs}_{\boldsymbol{T}} \end{bmatrix} \tag{4.15}$$

where $J_{\boldsymbol{E}}$ and $J_{\boldsymbol{T}}$ are the terms analogous to (4.13) for the EVSS and Tensor Stokes formulation. The details of the operators given in the above matrix structures can be found

in section A.2 of the appendix provided at the end. The impact of the stabilization can be observed in the comprehensive numerical study conducted in the subsequent chapters.

5

The numerical solver

This chapter focuses on the numerical solver for the discrete system resulting from the finite element discretization of partial differential equations. By discretizing the incompressible viscoelastic formulations, the system of nonlinear equations is obtained. The most commonly used approach for solving these equations involves a nonlinear iteration, with a linearized problem being solved at each step.

5.1 The outer non-linear solver

The job of the nonlinear solver turns out to be extremely critical with respect to the numerical stability. The fixed point and Newton method are the two available options for the outer solver. These two well-known linearization techniques are frequently used in numerical computations. The Newton method, which is well known for its quadratic convergence, if the initial solution is chosen close enough to the exact solution and the problem is smooth enough, is an efficient way to solve the nonlinear system. As a result, once the Newton method converges, it only requires a few iterations. However, due to the approximation of the Fréchet-derivative of the nonlinear operator, the construction of the Newton matrix in each nonlinear iteration might be quite expensive. Using the fixed point iteration approach, where the approximation of the Fréchet-derivative may be as chosen by the associated nonlinear operator itself or even the linear part of the nonlinear operator, is a more reliable option for solving the system of nonlinear algebraic equations. The fixed point method is more robust compared to the Newton method. Therefore, we have employed the fixed point method to solve the highly nonlinear problem.

5.2 The inner linear solver

The inner solver is intended to solve the linearized problem that is obtained in each nonlinear iteration. The linearized problem is of the form,

$$\mathcal{A}x = b \tag{5.1}$$

where the operator $\mathcal{A}$ is large and sparse. A sparse matrix is characterized by having a significant number of zero elements, enabling the development of specialized algorithms and data structures that optimize time and memory usage by exploiting these zeros [19]. It is a difficult task to find the solution of such large and sparse systems as it takes a lot of time, which raises the overall computational cost. The researchers have been actively working on the numerical techniques to efficiently solve the large and sparse linear systems. The linear solvers can be classified into two main categories i.e., direct solvers and iterative methods. When large computer memory is available, direct solvers for solving the sparse linear systems are preferred due to their accuracy and efficiency; however, if the computer memory is limited, iterative solvers are preferable. The choice between direct and iterative solvers depends on the balance between available memory, computational resources, and the desired level of accuracy.

5.2.1 Direct methods

Direct solvers, such as Gaussian elimination, compute the exact solution to a sparse linear system by performing a series of arithmetic operations on the matrix. It is the basic direct method for solving linear systems of equations. The smart variants of the Guass elimination method use sophisticated techniques to exploit the sparse matrix, lowering the computational cost. These methods includes LU decomposition, Frontal method, Multifrontal method for symmetric matrices and unsymmetric-pattern Multifrontal method which is also called UMFPACK. UMFPACK is the already available subroutine that is capable to solve unsymmetric, sparse linear systems [18, 1]. Direct methods are highly favorable for solving the sparse systems provided that high computer memory is available [20].

5.2.2 Iterative methods

Iterative methods are widely used in scientific computing to solve large and sparse linear systems as opposed to direct methods, because they require less memory, which reduces overall computational cost. Iterative methods for solving linear systems involve various techniques to find approximate solution through sequential approximations. Starting with

an initial guess, these methods iteratively refine the solution to achieve nearly accurate results. The iterative methods can be classified into three main categories namely, basic iterative methods, Krylov subspace methods and multigrid methods.

1. *Basic iterative methods*

   The class of basic iterative methods includes Jacobi method, Gauss-Seidel method and SOR (Successive over-relaxation method) etc. The Jacobi method, also referred as the method of simultaneous displacement, finds an approximate solution to a linear system of diagonally dominant equations. Each diagonal entry in the Jacobi method is filled with an approximate value. The iterative procedure is repeated until the algorithm converges. The Guass-Siedal method, also known as the method of successive displacement, is an iterative method that updates the current approximate solution with the newly computed approximate value. Successive Over Relaxation (SOR) is a variant of the Guass-Siedal method that results in faster convergence.

   On highly refined meshes, the basic iterative methods converge slowly, and the convergence rate is highly sensitive to the optimal choice of the involved damping parameter. As a result, the basic iterative methods are almost never used alone and are used as a preconditioners for advanced iterative solvers like Krylov subspace methods (Flexible Generalized Minimal Residual method FGMRES method) or as smoothers for multigrid methods [57].

2. *Flexible Generalized Minimal Residual method (FGMRES) method*

   Flexible Generalized Minimal Residual (FGMRES) belongs to the class of iterative solvers for linear systems. These solvers are essential for finding approximate solutions to large and sparse systems of linear equations, especially when direct methods like Gaussian elimination are impractical due to computational or memory constraints. FGMRES is an advanced class of iterative solver that utilize basic iterative solvers (Jacobi, Guass-Siedal, SOR) as preconditioners. Preconditioners play a crucial role in advanced iterative methods by helping to accelerate convergence and reduce computational effort when solving large linear systems. These preconditioners act as a sort of "warm-up" for the iterative solver, providing it with an initial guess that brings it closer to the solution. Hence, by integrating the basic iterative solvers as preconditioners into FGMRES, the solver becomes more effective at handling challenging linear systems, ultimately leading to faster and more accurate solutions.
   Other notable methods within this class include Conjugate Gradient (CG), which is primarily used for symmetric positive-definite matrices, and Bi-Conjugate Gradi-

ent (BiCG), which extends Conjugate Gradient to handle non-symmetric matrices. BiCGSTAB (Bi-Conjugate Gradient Stabilized) improves upon Bi-Conjugate Gradient by providing faster and more stable convergence for non-symmetric systems. GMRES (Generalized Minimal Residual) is suitable for non-symmetric and non-positive definite matrices, focusing on minimizing the residual over a Krylov subspace. FGMRES is a variant of the GMRES method, which is well-regarded for its robustness and versatility in handling a broad range of linear problems. The detailed description of these methods can be found in [56].

3. *Multigrid method*

The next class of the iterative methods is geometric multigrid method which is well known for being the most effective numerical method for solving the large system of linear algebraic equations. Employing geometric multigrid method for solving the linear system of equations proves appealing due to the potential for achieving mesh-independent convergence behavior. The method is based on solving the algebraic system using the hierarchy of grids with varied levels of resolution from fine to coarse. This iterative technique involves a combination of smoothing and grid transfer operations to efficiently converge to a solution. Therefore, the main components of the multigrid method comprises of the smoother, restriction, prolongation and a direct coarse grid solver (UMFPACK). Restriction is performed to convert the residual from finer mesh to coarser mesh. Considering the conforming finite element, the lower level space is a subspace of the finer level one. As a result, natural injection may be used for the restriction operator. The transfer from the coarse to the fine grid is called prolongation or interpolation. The pre and post smoothing steps are applied before and after the restriction and prolongation, respectively. The frequency with which the coarser mesh level is visited is determined by the cycle type, such as V-cycle, F-cycle, or W-cycle. The coarsest level is visited once in the V-cycle and several times in the F-cycle. The concept of multigrid solvers, as outlined in references [17, 64, 47, 67, 65], emphasizes the crucial role played by the chosen smoother within this framework.

5.3 Solution approaches

Having solved the challenges of the three-field viscoelastic formulation. The next step is to investigate solution techniques that can be used to solve the three-field viscoelastic formulation. The decoupling and fully monolithic method are two different approaches that can be used to obtain the solution of the three-field viscoelastic formulation. Each iteration of the former separates the computations of the Stokes problem (momentum and continuity

equation) from the constitutive equation. In the monolithic solution method, the three equations momentum, continuity and the constitutive equation are solved simultaneously in a fully coupled manner. The steady state decoupled and monolithic formulations are described below.

5.3.1 The decoupled / operator splitting method

In the decoupled or operator splitting approach, the three field formulation is split into two subproblems i.e. the Stokes part and the constitutive equation for stress. The velocity obtained from the Stokes part is employed in the constitutive equation and the corresponding stress obtained from the constitutive equation is updated in the momentum equation of the Stokes part in each iteration. We already know that the numerical solution of the viscoelastic formulation is quite expensive, such decoupled algorithm results in reducing the memory requirements and the CPU time. Let us begin with the non-stationary three field viscoelastic extra stress tensor formulation with zero solvent viscosity,

$$\begin{aligned} -\nabla \cdot \boldsymbol{S} + \nabla p &= 0 \\ \nabla \cdot \mathbf{u} &= 0 \\ \lambda \overset{\nabla}{\boldsymbol{S}} + f(\boldsymbol{S}, \lambda)\, \boldsymbol{S} + h\left(\boldsymbol{S}\right) &= 2\eta_p \mathbf{D}\left(\mathbf{u}\right) \end{aligned} \tag{5.2}$$

One could see no chance to solve the equation (5.2) in a decoupled manner in case of zero solvent viscosity. The absence of the diffusive term in the momentum equation makes it impossible to treat the Stokes and constitutive equation in the decoupled way. The decoupled approach for the extra stress tensor formulation with zero solvent viscosity could be solved with the help stabilization. The both-sides diffusion approach introduces additional diffusion terms on both sides of the momentum equation. The two additional terms are treated in the same manner in the discretization practice and, therefore equal a zero-addition in the numerical sense. This treatment helps to solve the three field problem in a decoupled manner with zero solvent viscosity. The resulting algorithm reads as,
Let $(\mathbf{u}^n, p^n, \boldsymbol{S}^n)$ be the known approximation of $(\mathbf{u}, p, \boldsymbol{S})$ after n steps. The step $(n+1)$ of the algorithm consists of first computing $(\mathbf{u}^{n+1}, p^{n+1})$ by solving the Stokes problem using the previous stress $\boldsymbol{S}^n$ in the right-hand side of the linear system, and then computing $\boldsymbol{S}^{n+1}$ by using its constitutive equation, with the new velocity $\mathbf{u}^{n+1}$ in the right-hand side. The iteration $(n+1)$ consists of first computing $(\mathbf{u}^{n+1}, p^{n+1})$ as follows,

$$\begin{cases} \nabla p^{n+1} - \beta \nabla \cdot \boldsymbol{D}\left(\mathbf{u}^{n+1}\right) = \nabla \cdot \boldsymbol{S}^n - \beta \nabla \cdot \mathbf{D}\left(\mathbf{u}^n\right) \\ \nabla \cdot \mathbf{u}^{n+1} = 0 \end{cases} \tag{5.3}$$

and in the second step we compute $\boldsymbol{S}^{n+1}$:

$$\lambda \overset{\nabla}{\boldsymbol{S}}^{n+1} + f(\boldsymbol{S}^n, \lambda)\, \boldsymbol{S}^{n+1} + h\left(\boldsymbol{S}^{n+1}\right) = 2\eta_p \mathbf{D}\left(\mathbf{u}^{n+1}\right) \tag{5.4}$$

where,

$$\overset{\nabla}{\boldsymbol{S}}^{n+1} = \left(\mathbf{u}^{n+1} \cdot \nabla\right) \boldsymbol{S}^{n+1} - \nabla \mathbf{u}^{T^{n+1}} \cdot \boldsymbol{S}^{n+1} - \boldsymbol{S}^{n+1} \cdot \nabla \mathbf{u}^{n+1} \tag{5.5}$$

The constant β controls the speed of the convergence and does not effect the solution. Thus, for any viscoelastic model one has to employ the additional stabilization terms when solving in a decoupled way. The stabilization parameter β could lead to problems at higher lambda values that will be discussed in the *Chapter 7*. We want to get rid of this dependency and make the algorithm independent of the any additional stabilization. Therefore, we have selected the EVSS formulation and the novel Tensor Stokes approach for the numerical computations. The EVSS and Tensor Stokes formulation helps us to obtain the meaningful solution using the decoupled approach even in the case of zero solvent viscosity.

Reconsider the steady-state EVSS formulation,

$$\begin{aligned} -2\eta_0 \nabla \cdot \mathbf{D}(\mathbf{u}) + \nabla p - \nabla \cdot \boldsymbol{E} &= 0 \\ \nabla \cdot \mathbf{u} &= 0 \\ \lambda(2\eta_p \mathbf{D}\overset{\nabla}{(\mathbf{u})} + \overset{\nabla}{\boldsymbol{E}}) + f(2\eta_p \mathbf{D}(\mathbf{u}) + \boldsymbol{E}, \lambda)\,(2\eta_p \mathbf{D}(\mathbf{u}) + \boldsymbol{E}) + & \\ h\,(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}) &= 2\eta_p \mathbf{D}(\mathbf{u}) \end{aligned} \tag{5.6}$$

The splitting of the extra stress tensor into the elastic and viscous part provides the opportunity to treat the three field viscoelastic formulation in a decoupled manner. The diffusive operator in the momentum equation has already been recovered and one does not need any additional stabilization. The decoupled algorithm for the EVSS formulation reads as,

Let $(\mathbf{u}^n, p^n, \boldsymbol{E}^n)$ be the known approximation of $(\mathbf{u}, p, \boldsymbol{E})$ after n steps. The step $(n+1)$ of the algorithm consists of first computing $(\mathbf{u}^{n+1}, p^{n+1})$ and then computing $\boldsymbol{E}^{n+1}$ by using its constitutive equation, with the new velocity $\mathbf{u}^{n+1}$. The iteration $(n+1)$ consists of first computing $(\mathbf{u}^{n+1}, p^{n+1})$ as follows,

$$\begin{aligned} -2\eta_0 \nabla \cdot \mathbf{D}\left(\mathbf{u}^{n+1}\right) + \nabla p^{n+1} - \nabla \cdot \boldsymbol{E}^n &= 0 \\ \nabla \cdot \mathbf{u}^{n+1} &= 0 \end{aligned} \tag{5.7}$$

In the second step compute $\boldsymbol{E}^{n+1}$

$$\begin{aligned} \lambda(2\eta_p \mathbf{D}\overset{\nabla}{\left(\mathbf{u}^{n+1}\right)} + \overset{\nabla}{\boldsymbol{E}^{n+1}}) + h\left(2\eta_p \mathbf{D}\left(\mathbf{u}^{n+1}\right) + \boldsymbol{E}^{n+1}\right) & \\ + f(2\eta_p \mathbf{D}(\mathbf{u}^n) + \boldsymbol{E}^n, \lambda)\left(2\eta_p \mathbf{D}\left(\mathbf{u}^{n+1}\right) + \boldsymbol{E}^{n+1}\right) &= 2\eta_p \mathbf{D}\left(\mathbf{u}^{n+1}\right) \end{aligned} \tag{5.8}$$

where,

$$\overset{\nabla}{\boldsymbol{E}}^{n+1} = \left(\mathbf{u}^{n+1}\cdot\nabla\right)\boldsymbol{E}^{n+1} - \nabla\mathbf{u}^{T^{n+1}}\cdot\boldsymbol{E}^{n+1} - \boldsymbol{E}^{n+1}\cdot\nabla\mathbf{u}^{n+1} \tag{5.9}$$

$$\overset{\nabla}{\mathbf{D}}^{n+1} = \left(\mathbf{u}^{n+1}\cdot\nabla\right)\mathbf{D}\left(\mathbf{u}^{n+1}\right) - \nabla\mathbf{u}^{T^{n+1}}\cdot\mathbf{D}\left(\mathbf{u}^{n+1}\right) - \mathbf{D}\left(\mathbf{u}^{n+1}\right)\cdot\nabla\mathbf{u}^{n+1} \tag{5.10}$$

Similarly, reconsider the steady-state Tensor Stokes formulation,

$$\begin{aligned} -\frac{1}{2}\nabla\cdot\left(\boldsymbol{T}\cdot\mathbf{D}\left(\mathbf{u}\right)+\mathbf{D}\left(\mathbf{u}\right)\cdot\boldsymbol{T}^{T}\right)+\nabla p &= 0 \\ \nabla\cdot\mathbf{u} &= 0 \\ \lambda(\overset{\nabla}{\boldsymbol{T}\cdot\mathbf{D}}) + f(\boldsymbol{T}\cdot\mathbf{D},\lambda)\left(\boldsymbol{T}\cdot\mathbf{D}\right) + h\left(\boldsymbol{T}\cdot\mathbf{D}\right) &= 2\eta_p\mathbf{D}\left(\mathbf{u}\right) \end{aligned} \tag{5.11}$$

One could see the recovery of the diffusive like operator in the momentum equation without any added stabilization. The presence of this operator helps in treating the set of equations in a decoupled way. The algorithm reads as, Let $(\mathbf{u}^n, p^n, \boldsymbol{T}^n)$ be the known approximation of $(\mathbf{u}, p, \boldsymbol{T})$ after n steps. The step $(n+1)$ of the algorithm consists of first computing $(\mathbf{u}^{n+1}, p^{n+1})$ by using some constant value of $\boldsymbol{T}$, and then computing $\boldsymbol{T}^{n+1}$ by using its constitutive equation, with the new velocity $\mathbf{u}^{n+1}$. The iteration $(n+1)$ consists of first computing $(\mathbf{u}^{n+1}, p^{n+1})$ as follows:

$$\begin{aligned} -\frac{1}{2}\nabla\cdot\left(\boldsymbol{T}^{n}\cdot\mathbf{D}\left(\mathbf{u}^{n+1}\right)+\mathbf{D}\left(\mathbf{u}^{n+1}\right)\cdot\boldsymbol{T}^{T^{n}}\right)+\nabla p^{n+1} &= 0 \\ \nabla\cdot\mathbf{u}^{n+1} &= 0 \end{aligned} \tag{5.12}$$

In the second step we compute $\boldsymbol{T}^{n+1}$,

$$\begin{aligned} &\lambda\big(\overset{\nabla}{\boldsymbol{T}^{n+1}\cdot\mathbf{D}^{n+1}}\big) + h\left(\boldsymbol{T}^{n+1}\cdot\mathbf{D}^{n+1}\right) \\ &\quad + f\big(\boldsymbol{T}^{n}\cdot\mathbf{D}^{n+1},\lambda\big)\left(\boldsymbol{T}^{n+1}\cdot\mathbf{D}^{n+1}\right) = 2\eta_p\mathbf{D}\left(\mathbf{u}^{n+1}\right) \end{aligned} \tag{5.13}$$

where,

$$\begin{aligned} \big(\overset{\nabla}{\boldsymbol{T}^{n+1}\cdot\mathbf{D}^{n+1}}\big) &= \left(\mathbf{u}^{n+1}\cdot\nabla\right)\left(\boldsymbol{T}^{n+1}\cdot\mathbf{D}^{n+1}\right) - \\ \Big(\nabla\mathbf{u}^{T^{n+1}}\cdot\left(\boldsymbol{T}^{n+1}\cdot\mathbf{D}^{n+1}\right) &+ \left(\boldsymbol{T}^{n+1}\cdot\mathbf{D}^{n+1}\right)\cdot\nabla\mathbf{u}^{n+1}\Big) \end{aligned} \tag{5.14}$$

Thus, the well known EVSS formulation and the novel Tensor Stokes formulation offers the possibility to treat complex fluids in a viscoelastic framework in a decoupled manner. That means one has to deal with the smaller subproblems that in turn helps to reduce the computational cost.

5.3.2 Monolithic approach

The solution of the three field viscoelastic formulations can be obtained in a fully monolithic manner i.e the three equations, momentum, continuity and the constitutive equation

are solved simultaneously. Such numerical handling refers to the velocity-pressure-stress formulation. The V-P-S formulation that solves all unknowns simultaneously leads to a robust and stable numerical approach. The EVSS and the Tensor Stokes formulations provide the opportunity to handle the three field viscoelastic formulations in a decoupled as well as in a fully monolithic way.
Reconsider system of equations in (5.2), the monolithic treatment to solve such system is given in (5.15). Let $(\mathbf{u}^n, p^n, \boldsymbol{S}^n)$ be the known approximation of $(\mathbf{u}, p, \boldsymbol{S})$ after n steps. The step $(n+1)$ of the algorithm consists of computing $(\mathbf{u}^{n+1}, p^{n+1}, \boldsymbol{S}^{n+1})$ as follows:

$$\begin{aligned} \nabla p^{n+1} - \nabla \cdot \boldsymbol{S}^{n+1} &= 0 \\ \nabla \cdot \mathbf{u}^{n+1} &= 0 \\ \lambda \overset{\nabla}{\boldsymbol{S}}^{n+1} + f(\boldsymbol{S}^n, \lambda)\, \boldsymbol{S}^{n+1} + h\big(\boldsymbol{S}^{n+1}\big) - 2\eta_p \mathbf{D}\left(\mathbf{u}^{n+1}\right) &= 0 \end{aligned} \tag{5.15}$$

where,

$$\overset{\nabla}{\boldsymbol{S}}^{n+1} = \left(\mathbf{u}^n \cdot \nabla\right) \boldsymbol{S}^{n+1} - \nabla \mathbf{u}^{T^n} \cdot \boldsymbol{S}^{n+1} - \boldsymbol{S}^{n+1} \cdot \nabla \mathbf{u}^n \tag{5.16}$$

It can be observed that the zero solvent viscosity results in the zero block of the Stokes matrix in the main diagonal. The absence of the diffusive operator poses difficulties in obtaining the solution even in a monolithic way. The EVSS and the Tensor Stokes formulation provides the opportunity to recover the diffusive operator in the momentum equation even in case of zero solvent viscosity.

The monolithic treatment of the EVSS formulation is given in (5.17). Let $(\mathbf{u}^n, p^n, \boldsymbol{E}^n)$ be the known approximation of $(\mathbf{u}, p, \boldsymbol{E})$ after n steps. The step $(n+1)$ of the algorithm consists of computing $(\mathbf{u}^{n+1}, p^{n+1}, \boldsymbol{E}^{n+1})$ as follows:

$$\begin{aligned} -2\eta_0 \nabla \cdot \mathbf{D}\left(\mathbf{u}^{n+1}\right) + \nabla p^{n+1} - \nabla \cdot \boldsymbol{E}^{n+1} &= 0 \\ \nabla \cdot \mathbf{u}^{n+1} &= 0 \\ \lambda(2\eta_p \overset{\nabla}{\mathbf{D}}\left(\mathbf{u}^{n+1}\right) + \overset{\nabla}{\boldsymbol{E}}^{n+1}) + h\left(2\eta_p \mathbf{D}\left(\mathbf{u}^{n+1}\right) + \boldsymbol{E}^{n+1}\right) & \\ + f(2\eta_p \mathbf{D}\left(\mathbf{u}^n\right) + \boldsymbol{E}^n, \lambda)\left(2\eta_p \mathbf{D}\left(\mathbf{u}^{n+1}\right) + \boldsymbol{E}^{n+1}\right) - 2\eta_p \mathbf{D}\left(\mathbf{u}^{n+1}\right) &= 0 \end{aligned} \tag{5.17}$$

where,

$$\overset{\nabla}{\boldsymbol{E}}^{n+1} = \left(\mathbf{u}^n \cdot \nabla\right) \boldsymbol{E}^{n+1} - \nabla \mathbf{u}^{T^n} \cdot \boldsymbol{E}^{n+1} - \boldsymbol{E}^{n+1} \cdot \nabla \mathbf{u}^n \tag{5.18}$$

$$\overset{\nabla}{\mathbf{D}}^{n+1} = \left(\mathbf{u}^n \cdot \nabla\right) \mathbf{D}\left(\mathbf{u}^{n+1}\right) - \nabla \mathbf{u}^{T^n} \cdot \mathbf{D}\left(\mathbf{u}^{n+1}\right) - \mathbf{D}\left(\mathbf{u}^{n+1}\right) \cdot \nabla \mathbf{u}^n \tag{5.19}$$

The monolithic treatment of the Tensor Stokes formulation (5.11) is given in (5.20). Let $(\mathbf{u}^n, p^n, \boldsymbol{T}^n)$ be the known approximation of $(\mathbf{u}, p, \boldsymbol{T})$ after n steps. The step $(n+1)$ of the algorithm consists of computing $(\mathbf{u}^{n+1}, p^{n+1}, \boldsymbol{T}^{n+1})$ as follows:

$$
\begin{aligned}
-\frac{1}{2}\nabla \cdot \left(\boldsymbol{T}^{n} \cdot \mathbf{D}\left(\mathbf{u}^{n+1}\right) + \mathbf{D}\left(\mathbf{u}^{n+1}\right) \cdot \boldsymbol{T}^{T^{n}}\right) + \nabla p^{n+1} &= 0 \\
\nabla \cdot \mathbf{u}^{n+1} &= 0 \\
\lambda\big(\boldsymbol{T}^{n+1} \overset{\nabla}{\cdot} \mathbf{D}(\mathbf{u}^{n})\big) + f(\boldsymbol{T}^{n} \cdot \mathbf{D}(\mathbf{u}^{n}), \lambda)\left(\boldsymbol{T}^{n+1} \cdot \mathbf{D}(\mathbf{u}^{n})\right) + & \\
h\left(\boldsymbol{T}^{n+1} \cdot \mathbf{D}(\mathbf{u}^{n})\right) - 2\eta_{p}\mathbf{D}\left(\mathbf{u}^{n+1}\right) &= 0
\end{aligned}
\tag{5.20}
$$

where,

$$
\begin{aligned}
\big(\boldsymbol{T}^{n+1} \overset{\nabla}{\cdot} \mathbf{D}(\mathbf{u}^{n})\big) = (\mathbf{u}^{n} \cdot \nabla)\left(\boldsymbol{T}^{n+1} \cdot \mathbf{D}(\mathbf{u}^{n})\right) - & \\
\left(\nabla \mathbf{u}^{T^{n}} \cdot \left(\boldsymbol{T}^{n+1} \cdot \mathbf{D}(\mathbf{u}^{n})\right) + \left(\boldsymbol{T}^{n+1} \cdot \mathbf{D}(\mathbf{u}^{n})\right) \cdot \nabla \mathbf{u}^{n}\right) &
\end{aligned}
\tag{5.21}
$$

The recovery of the diffusive operator in the momentum equation of the EVSS and the Tensor Stokes formulation leads to non-zero block on the diagonal of the Stokes part. Therefore, the considered viscoelastic formulations not only aid in the decoupled solution approach but also plays an important role in obtaining the solutions in a monolithic way. Having discussed the numerical solver and solution strategies in detail, let us proceed towards the numerical results in the next chapters.

6

Validation of the code

This chapter is dedicated to the validation of the code, which is a very crucial part of the thesis. The numerical simulations are performed on the compute servers of TU Dortmund University using the FEATFLOW3 software, which is a novel C++ code with a highly adaptable solver structure, that is intended for usage by researchers as well as in industry applications. The link to the code can be found on the GitHub page [1]. The first application is the stationary Newtonian three field Stokes formulation tested on the flow around the cylinder benchmark, where drag and lift values are computed and validated with the available literature. The numerical tests are also performed on a interesting flow geometry i.e., the stick-slip transitions. The viscoelastic results are also part of this chapter, where error between the analytic and numerical solutions are discussed for increasing values of the relaxation parameter λ.

6.1 Flow around the cylinder benchmark

The flow around cylinder benchmark was initiated by Turek and Schäfer [63] within the DFG high-priority research program "*flow simulation with high performance computers*". The underlying 2D geometry is a pipe without a circular cylinder $\Omega = (0, 2.2) \times (0, 0.41) / B_r\,(0.2, 0.2)$ as shown in Fig. 6.1.

[1] https://github.com/tudo-math-ls3/feat3

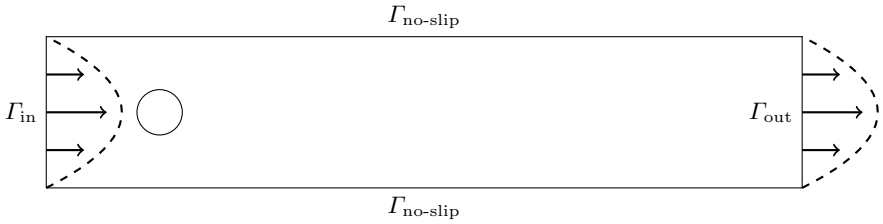


Fig. 6.1: Configuration for the flow around the cylinder

In the above configuration, a parabolic velocity profile given in equation (6.1) is prescribed at the inflow $\Gamma_{in} = 0 \times [0, 0.41]$, of the following form:

$$U(0, y) = \left(\frac{(4Uy(0.41 - y)}{0.41^2}, 0 \right) \tag{6.1}$$

with a maximum velocity $U = 0.3$. The no-slip boundary conditions are prescribed at the lower and upper wall as well as the boundary $S = \partial B_r(0.2, 0.2)$.

$$u_{|\Gamma_{\text{no-slip}}} = u_{|S} = 0 \tag{6.2}$$

Do nothing boundary condition are set at the outflow $\Gamma_{out} = 2.2 \times [0, 0.41]$. The simulations are performed using UMFPACK solver for the decoupled stationary three field Newtonian Stokes flow (i.e., $\lambda = 0$) in the extra stress tensor formulation given in equation (5.3) and (5.4). The convergence is ensured, when the norm of the defect vectors meets a specified tolerance. In order to validate the results obtained from three field Stokes flow we computed the drag and lift values around the surface of the cylinder and compare the results obtained by [17]. The drag and lift forces are given as follows:

$$C_{drag/lift} = C \int_s \boldsymbol{S} \cdot \mathbf{n} \; ds \tag{6.3}$$

where $C = {}^{2F_{d/l}}/_{\rho u_c^2 l_c}$ is a constant. Fig. 6.2 represents the coarse mesh for the flow around the cylinder with 130 quadrilaterals. We have performed the numerical computations us-

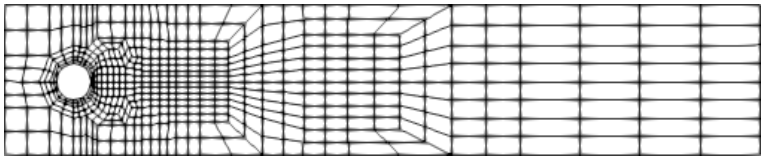


Fig. 6.2: Flow around a cylinder coarse mesh having 130 quadrilaterals.

ing two different finite element triplets for our standard extra-stress tensor formulation.

The two finite element discretizations of $(\mathbf{u}, p, \boldsymbol{S})$ are $\left(Q_2, P_1^{disc}, Q_2\right)$ and $\left(Q_2, P_1^{disc}, Q_3\right)$. Table 6.1 represents the drag and the lift values for the three field formulation obtained by using two different finite elements triplets mentioned above. It is clear from Table 6.1 that our resuts are in good agreement with the reference results obtained by [17].

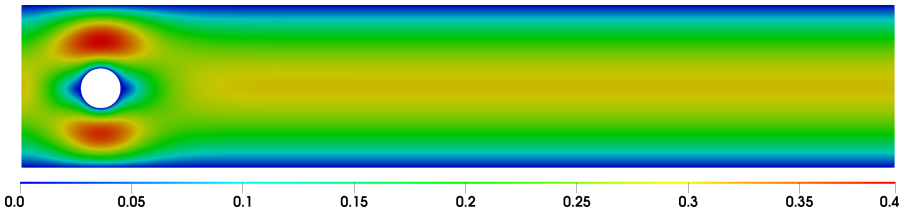


Fig. 6.3: **Flow around the cylinder:** Solution of $x-$velocity at mesh refinement level 3.

Level	Flow around the cylinder					
	(Q_2, P_1^{disc}, Q_2)		(Q_2, P_1^{disc}, Q_3)		Benchmark results	
	Drag	Lift	Drag	Lift	Drag	Lift
1	3.09321	0.0297908	3.090441	0.02983339	3.112646	0.02965870
2	3.13075	0.0301286	3.129198	0.03011847	3.134342	0.03005275
3	3.13960	0.0301817	3.139104	0.03017739	3.140327	0.03015909
4	3.14173	0.0301925	3.141596	0.03019148	3.141893	0.03018665

Table 6.1: **Newtonian flow:** Drag and lift values using two different finite element triplets, and compared with the reference results computed by [17] for different mesh refinement levels.

It can be observed that the present results computed using two different approximations for stress are in good agreement with the benchmark results obtained by [17]. In literature it has been accepted that the stable finite element pair for the monolithic formulation requires that the stress space should be larger than the velocity space [42]. We have analyzed the effect of bicubic approximation Q_3 for stress on the convergence behavior for the decoupled formulation in Table 6.2. Using the equal order discretization for the velocity and the stress i.e., both velocity and stress are approximated using Q_2 leads to very high number of iterations required for convergence as compared to the stress being discretized in Q_3 finite element. Keeping in view the criteria for the stable finite element triplet for the three field monolithic formulation and supremacy of the higher order approximation i.e., Q_3 for stress in terms of convergence we have selected bicubic

	Number of iterations	
Level	(Q_2, P_1^{disc}, Q_2)	(Q_2, P_1^{disc}, Q_3)
1	541	34
2	1367	44
3	2213	37
4	2308	21

Table 6.2: **Biquadratic vs bicubic discretization:** Number of iterations required for convergence using biquadratic and bicubic discretizations for stress at different mesh refinement levels.

finite element for stress for numerical computations in the subsequent sections. The next section includes results for the stick-slip application.

6.2 Stick-slip problem

In this subsection, we have considered stick-slip flow problem. The difficulty in the stick slip problem comes from the transition from a Dirichlet boundary condition (stick) to a Neumann boundary condition (slip). The point where this transition occurs is singular even in the Newtonian case, and it is extremely difficult to approximate the stresses in the vicinity of that point. Similar to the flow around the cylinder benchmark, we extend our numerical computations using UMFPACK solver for the computation of the results using $\left(Q_2, P_1^{disc}, Q_3\right)$ finite element triplet for $(\mathbf{u}, p, \boldsymbol{S})$ on the the stick-slip geometry. The convergence is ensured when the norm of the defect vectors meets a specified tolerance of 10^{-6}. The underlying geometry shown in Fig.6.4 is a rectangular domain $\Omega = (-20, +50) \times (-1, +1)$.

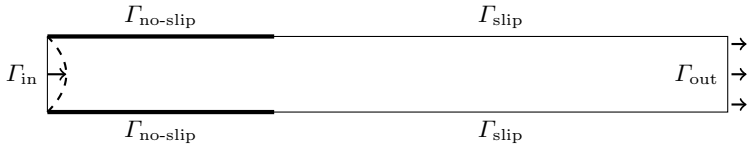


Fig. 6.4: Configuration for the stick-slip problem.

Boundary conditions:

- The lower wall is divided into two parts, $\Gamma_{\text{no-slip}} = [-20, 0] \times -1$ and $\Gamma_{\text{slip}} = (0, +50] \times -1$. On $\Gamma_{\text{no-slip}}$ no slip boundary conditions are applied and on Γ_{slip}, slip boundary conditions are applied.
- The upper wall is also divided into two parts, $\Gamma_{\text{no-slip}} = [-20, 0] \times 1$ and $\Gamma_{\text{slip}} = (0, +50] \times 1$. On $\Gamma_{\text{no-slip}}$, no slip boundary conditions are applied and, on Γ_{slip}, slip boundary conditions are applied.
- On the left edge, $\Gamma_{\text{in}} = -20 \times [-1, +1]$, a parabolic inflow profile is prescribed,

$$U(0, y) = (Uy(2 - y), 0),$$

  with a maximum velocity $U = 0.3$.
- On the right edge, at the outflow, $\Gamma_{\text{out}} = +50 \times [-1, +1]$, do nothing boundary conditions are prescribed.

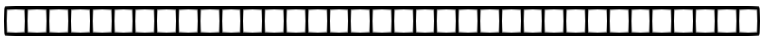


Fig. 6.5: Coarse mesh for the stick slip problem having 35 quadrilaterals.

Fig. 6.5 shows the coarse mesh with 35 quadrilaterls. The stable finite element pair developed by Marchal and Crochet was tested for the three field Newtonian Stokes flow over a stick-slip geometry [42]. They employed the nine-node Lagrangian elements with biquadratic shape functions for the velocity and the bilinear shape functions for the pressure. For the stress, they proposed a subcell discretization to enrich the local degrees of freedom. They divided the quadrilateral into n^2 bilinear sub-elements, where n can be 2, 3 and 4 respectively. The results obtained by Marchal and Crochet [42] can be seen in Fig. 6.8. Marchal and Crochet concluded that the use of bilinear sub-elements specially *four* stress sub-elements appears to be a very good idea in order to improve the convergence properties of the mixed method.
Keeping in view the success of the element proposed by Marchal and Crochet we have tested the selected finite element triplet $\left(Q_2, P_1^{disc}, Q_3\right)$ for $(\mathbf{u}, p, \boldsymbol{S})$, for the three field Newtonian flow. The x-velocity profile for the stick slip problem can be seen in Fig.6.6 where one could see the transition of flow from the *stick* region to *slip* region. Fig.6.7 is prepared to show the velocity and stress components exactly at the transition point from *stick* to *slip*. It can be observed that the present results computed using Q_3 finite element for stress are in good agreement with the results computed by Marchal and Crochet in Fig. 6.8 using 3×3 and 4×4 stress subelements.

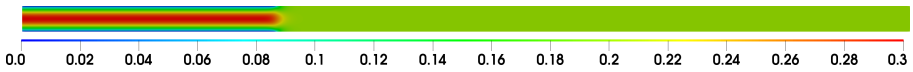


Fig. 6.6: **Stick-slip problem:** Solution of x-velocity at mesh refinement level 3.

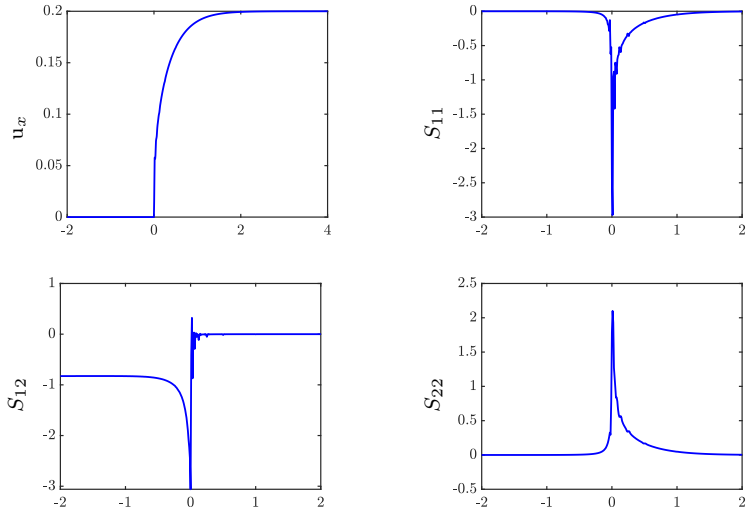


Fig. 6.7: Newtonian solution of the stick-slip problem along the line $x = 1$ obtained by using $\left(Q_2, P_1^{disc}, Q_3\right)$ finite element triplet for $(\mathbf{u}, p, \boldsymbol{S})$.

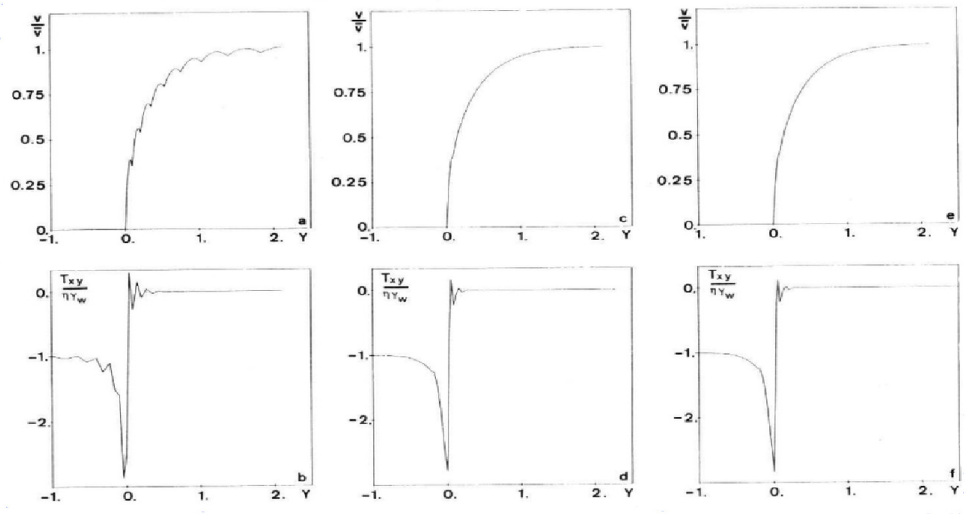


Fig. 6.8: Newtonian solution of the stick-slip problem along the line x=1 obtained by Marchal and Crochet [42] using 2×2, 3×3 and 4×4 sub-elements.

Having tested the Newtonian three field Stokes formulation with extra stress tensor formulation on the flow around the cylinder and stick slip problem, the numerical results for the simplest viscoelastic Oldroyd-B fluid using extra-stress tensor, EVSS and Tensor Stokes formulation are presented in the next section.

6.3 Steady state extra-stress tensor formulation for Oldroyd-B fluid

The three field steady state extra-stress tensor viscoelastic formulation is tested for the Shear and Poiseuille flow for the simplest known viscoelastic model i.e., Oldroyd-B. The simulations are performed using the decoupled (5.3) and (5.4) as well as the monolithic formulation given in (5.15). The Oldroyd-B model is selected because of the availability of the analytic functions for the velocity and stress. The error between the analytic and numerical results are reported for increasing values of the relaxation parameter λ.

6.3.1 Application to the shear flow

Consider a situation where two parallel plates are moving in opposite directions, resulting in a shear flow between them. The prescribed velocity on the inflow Γ_1 and outflow Γ_2 edges ensures a consistent shear rate across the flow domain. The underlying geometry shown in Fig. 6.9 is a square domain $\Omega \in [0, 2] \times [-1, 1]$.

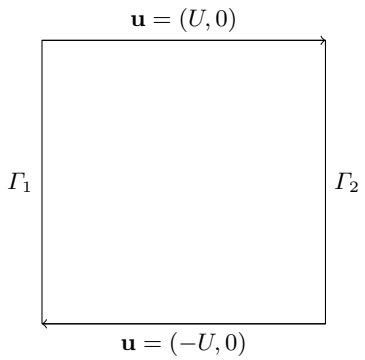


Fig. 6.9: Configuration for the shear flow

Boundary conditions:

- A velocity profile $\mathbf{u} = (Uy, 0)$ is prescribed on the inflow and outflow edges.
- The velocity $\mathbf{u} = (-U, 0)$ is set for the lower wall, and $\mathbf{u} = (U, 0)$ for the upper wall.
- The stress profile is specified only at the inflow.

The numerical simulations are performed for increasing values of the relaxation parameter λ on mesh refinement level 6 with 4096 quadrilaterals. The UMFPACK solver is employed to solve both the Stokes problem and constitutive equation for stress. The analytic velocity and stress functions for the Oldroyd-B fluid are given below,

$$\mathbf{u} = \begin{pmatrix} Uy \\ 0 \end{pmatrix}, \quad \boldsymbol{S} = \begin{pmatrix} 2\lambda\eta_p U^2 & \eta_p U \\ \eta_p U & 0 \end{pmatrix} \tag{6.4}$$

where $\eta_p = 1$ and $U = 0.1$. The error for the Oldroyd-B fluid is reported for the decoupled and monolithic formulation between the exact and numerical results as shown in Table 6.3. It can be observed that the numerical results obtained using the decoupled as well as the monolithic formulation are in good match with the analytic one for increasing values of the relaxation parameter λ. However, no convergence could be achieved for relaxation parameter value $\lambda > 0.1$.

	Application to Shear flow			
λ	$\|\|\mathbf{u}_{analytic} - \mathbf{u}\|\|$		$\|\|\boldsymbol{S}_{analytic} - \boldsymbol{S}\|\|$	
	Decoupled	Monolithic	Decoupled	Monolithic
0.000	3.412195e-15	4.261473e-14	1.123517e-12	1.561826e-12
0.001	1.197372e-14	3.431098e-15	1.084630e-11	1.104655e-12
0.010	6.369624e-15	3.590923e-15	6.918762e-13	9.485555e-13
0.100	2.832293e-14	4.314157e-15	8.753023e-12	1.417732e-12

Table 6.3: **Shear flow for Oldroyd-B fluid:** Error results for the Oldroyd-B Shear flow for increasing values of the relaxation parameter λ at mesh refinement level 6.

6.3.2 Application to the Poiseuille flow

In Poiseuille flow, the fluid moves between the parallel plates, with faster-moving fluid particles near the center of the flow and slower-moving particles near the walls, resulting in a parabolic velocity profile. The underlying geometry shown in Fig. 6.10 is a square domain $\Omega = (0, +1) \times (0, +1)$.

Boundary conditions:

- On the left and right edge prescribe a velocity profile $\mathbf{u} = (U\,(1-y^2)\,,0)$
- On the lower and upper wall set no slip boundary conditions
- The stress profile is prescribed only at the inflow.

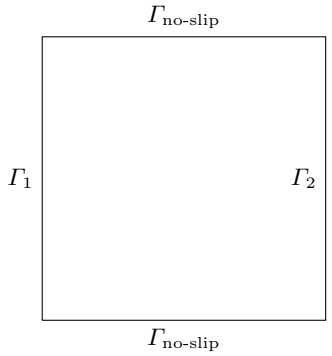


Fig. 6.10: Configuration for the Poiseuille flow.

Similar to the shear flow, the numerical computations are performed for the Poiseuille flow for increasing values of the relaxation parameter λ on mesh refinement level 6 with 4096 quadrilaterals. The UMFPACK solver is employed to solve both the Stokes problem and constitutive equation for stress. The analytic velocity and stress functions for the Oldroyd-B fluid are given follows:

$$\mathbf{u} = \begin{pmatrix} U\,(1-y^2) \\ 0 \end{pmatrix}, \quad \boldsymbol{S} = \begin{pmatrix} 8\lambda\eta_p U^2 y^2 & -2\eta_p U y \\ -2\eta_p U y & 0 \end{pmatrix} \tag{6.5}$$

where $\eta_p = 1$ and $U = 0.1$.

The comparison of error results for the Oldroyd-B fluid between the exact and numerical results is presented in Table 6.4 for both the decoupled and monolithic formulations. It can be seen that for increasing values of the relaxation parameter λ, the numerical results obtained from both formulations closely matches the analytic solution. For the Poiseuille flow, we could only achieve convergence until relaxation parameter $\lambda = 0.1$.

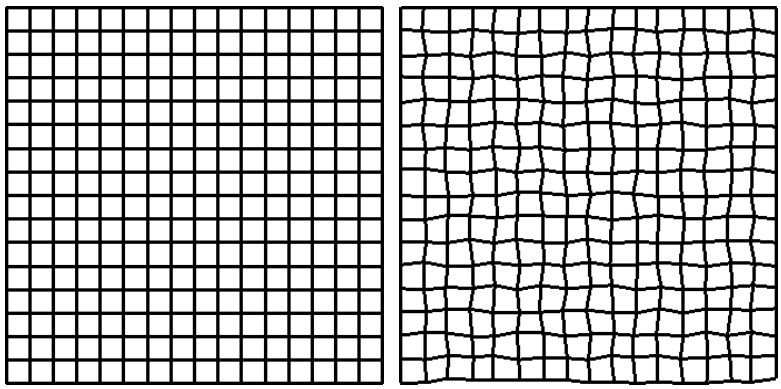
	Application to Poiseuille flow			
λ	$\|\|\mathbf{u}_{analytic} - \mathbf{u}\|\|$		$\|\|\boldsymbol{S}_{analytic} - \boldsymbol{S}\|\|$	
	Decoupled	Monolithic	Decoupled	Monolithic
0.0000	3.412195e-15	4.261473e-14	1.123517e-12	1.561826e-12
0.0010	1.197372e-14	3.431098e-15	1.084630e-11	1.104655e-12
0.0100	6.369624e-15	3.590923e-15	6.918762e-13	9.485555e-13
0.1000	2.832293e-14	4.314157e-15	8.753023e-12	1.417732e-12

Table 6.4: **Poiseuille flow for Oldroyd-B fluid:** Error for the Oldroyd-B Poiseuille flow for increasing values of the relaxation parameter λ at mesh refinement level 6.

The numerical results obtained from the decoupled and monolithic formulation are in good agreement to each other as well as the analytic functions for both the shear and Poiseuille flow for the extra-stress tensor formulation.
In the next sections, the numerical results for the simplest Oldroyd-B fluid are obtained for the EVSS and Tensor Stokes formulation for the Poiseuille flow on two different types of meshes, uniform and perturbed mesh.

6.4 Oldroyd-B results for EVSS and Tensor Stokes on uniform and perturbed mesh

In this section, the EVSS and Tensor Stokes three field viscoelastic formulations are validated against the analytic functions for the simplest Oldroyd-B model. The numerical tests are performed for both the decoupled and monolithic formulation on the square domain, utilizing two types of meshing: uniform and perturbed mesh. The perturbed mesh introduces a 0.1% perturbation, which is not aligned along $y = 0$. Fig. 6.11 is a square domain $\Omega \in [0, 2] \times [-1, 1]$ that illustrates the uniform and perturbed mesh configurations, each consisting of 16×16 quadrilaterals. A general quadrilateral mesh is generated by perturbing the uniform mesh, shifting the nodes by 10%. The numerical computations are performed on mesh refinement level 6 with 4096 quadrilaterals using UMFPACK solver.



(a) Uniform mesh

(b) Perturbed mesh not aligned along $y = 0$

Fig. 6.11: Square domain with 16×16 quadrilaterals

6.4.1 EVSS formulation: Steady state Oldroyd-B fluid flow

The steady state EVSS formulation obtained from the decomposition of the extra stress tensor into elastic and viscous part is tested for the simplest Oldroyd-B model on the square domain. The numerical computations are performed using the decoupled (5.7) and (5.8) and monolithic formulation (5.17). The elastic part of the extra stress tensor i.e., the stress component for EVSS formulation $\boldsymbol{E}$ is obtained using the extra stress function $\boldsymbol{S}$ in equation (6.5). The problem is subject to the following boundary conditions,

- A velocity profile $\mathbf{u} = (U\left(1 - y^2\right), 0)$ is prescribed on the inflow and outflow edges
- For the uniform mesh, set no slip boundary condition on the lower and upper wall
- For the perturbed mesh, set no slip boundary condition on the upper wall and symmetry boundary condition on the lower wall
- The stress profile is specified only at the inflow.

The analytic velocity and stress functions for the Oldroyd-B fluid are given below,

$$\mathbf{u} = \begin{pmatrix} U\left(1 - y^2\right) \\ 0 \end{pmatrix}, \quad \boldsymbol{E} = \begin{pmatrix} 8\lambda\eta_p U^2 y^2 & 0 \\ 0 & 0 \end{pmatrix} \tag{6.6}$$

where $\eta_p = 1$ and $U = 0.1$. Table 6.5 shows the error results between the analytic and numerical results for increasing values of the relaxation parameter λ performed on uniform and perturbed mesh obtained via decoupled formulation. The similar error computation is performed for the monolithic formulation given in Table 6.6. It is worth nothing that both the formulations help to achieve results at quite high values of the relaxation parameter λ i.e., $\lambda = 10$.

	EVSS decoupled formulation			
λ	$\|\|\mathbf{u}_{analytic} - \mathbf{u}\|\|$		$\|\|\boldsymbol{E}_{analytic} - \boldsymbol{E}\|\|$	
	Uniform	Perturbed	Uniform	Perturbed
0.0	3.714974e-15	2.122570e-15	0.00000e+00	0.00000e+00
0.1	8.101435e-13	2.750346e-14	1.794326e-11	1.770841e-11
1.0	1.968628e-13	6.128845e-14	2.588079e-11	3.267391e-11
2.0	2.751582e-13	8.698643e-14	3.261383e-11	4.774626e-11
5.0	7.087514e-13	1.089180e-13	5.653581e-11	5.178690e-11
10	1.084389e-13	1.314691e-13	8.444790e-11	6.014599e-11

Table 6.5: **EVSS decoupled Oldroyd-B fluid:** Error results for the Oldroyd-B flow for increasing values of the relaxation parameter λ obtained via decoupled formulation

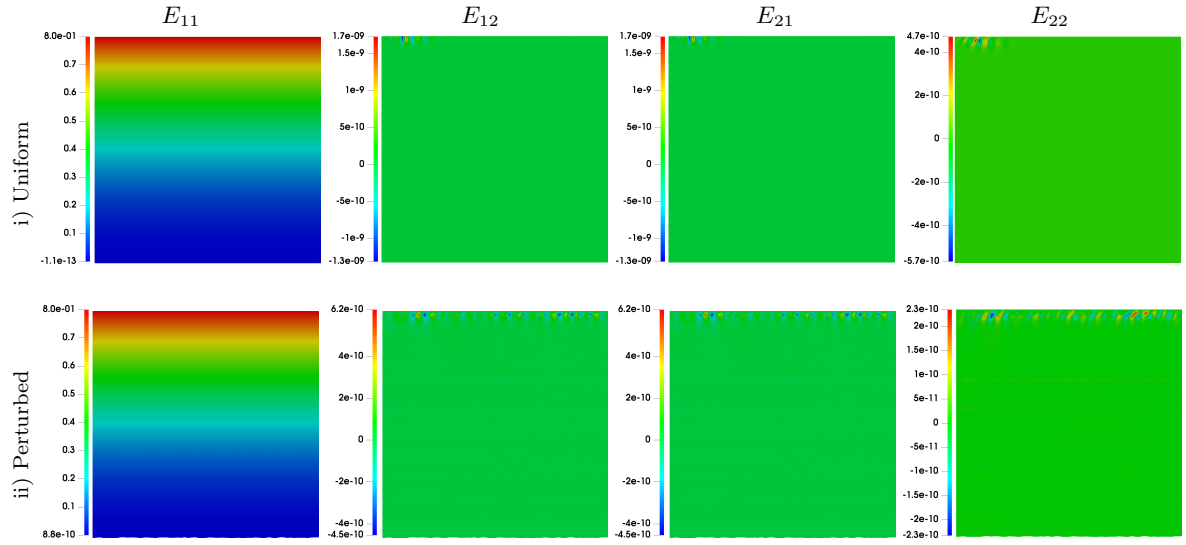


Fig. 6.12: **Decoupled approach for Oldroyd-B fluid:** Components of the stress tensor $\boldsymbol{E}$ obtained via decoupled formulation on uniform and perturbed mesh, for the relaxation parameter $\lambda = 10$ at mesh refinement level 6.

In Fig. 6.12 and 6.13 we are going to display components of the tensor $\boldsymbol{E}$ on the uniform and perturbed mesh obtained via decoupled and monolithic formulation, respectively. Observing component E_{11} in both figures, it is evident that it closely resembles the analytic function described in equation (6.6). The stress reaches its peak near the top wall at $y = 1$ and its minimum value at the lower wall at $y = 0$. Whereas, all remaining components of stess, E_{12}, E_{21} and E_{22} are zero inside the flow domain.

	EVSS monolithic formulation			
λ	$\|\|\mathbf{u}_{analytic} - \mathbf{u}\|\|$		$\|\|\boldsymbol{E}_{analytic} - \boldsymbol{E}\|\|$	
	Uniform	Perturbed	Uniform	Perturbed
0.0	2.958058e-15	4.261473e-14	7.312293e-13	1.561826e-12
0.1	9.743262e-15	1.868345e-14	3.585552e-13	3.572130e-12
1.0	1.197372e-14	1.403583e-14	1.084630e-11	5.439474e-12
2.0	6.369624e-15	2.189742e-14	6.918762e-13	7.262114e-12
5.0	2.832293e-14	7.151332e-15	8.753023e-12	5.027944e-12
10.0	6.815502e-15	8.194507e-15	6.535207e-12	5.987757e-12

Table 6.6: **EVSS monolithic Oldroyd-B fluid:** Error computations for the Oldroyd-B flow for increasing values of the relaxation parameter λ obtained via monolithic formulation

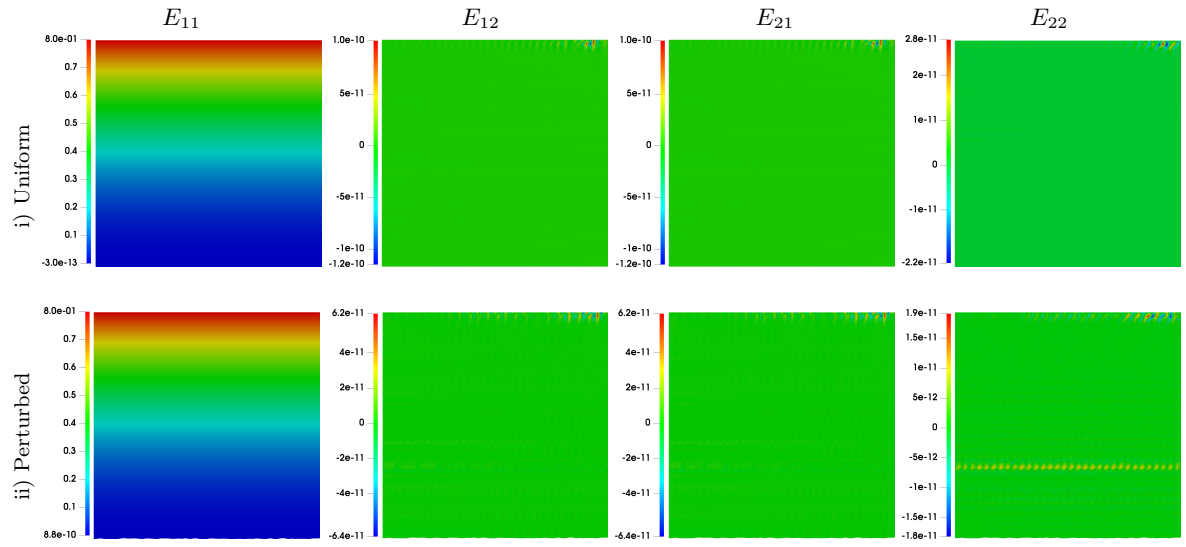


Fig. 6.13: **Monolithic approach for Oldroyd-B fluid:** Components of the stress tensor $\boldsymbol{E}$ obtained via monolithic formulation on uniform and perturbed mesh, for the relaxation parameter $\lambda = 10$ at mesh refinement level 6.

6.4.2 Tensor Stokes formulation: Steady state Oldroyd-B fluid flow

The steady-state Tensor Stokes formulation, derived by decomposing the extra stress tensor into the product of tensor based viscosity and deformation tensor, is evaluated for

the simplest Oldroyd-B model on a square domain given in Fig. 6.11. Numerical tests are performed using both the decoupled equations (5.12) and (5.13), as well as the monolithic formulation (5.20). The viscosity tensor, denoted as $\boldsymbol{T}$ in the Tensor Stokes formulation, is derived from the extra stress function $\boldsymbol{S}$ as defined in equation (6.5). The problem is governed by the following boundary conditions:

- A velocity profile $\mathbf{u} = (U\left(1-y^2\right), 0)$ is prescribed on the inflow and outflow edges
- For the uniform mesh, set no slip boundary condition on the lower and upper wall
- For the perturbed mesh, set no slip boundary condition on the upper wall and symmetry boundary condition on the lower wall
- The stress profile is specified only at the inflow.

The analytic velocity and stress functions for the Oldroyd-B fluid are given below,

$$\mathbf{u} = \begin{pmatrix} U\left(1-y^2\right) \\ 0 \end{pmatrix}, \quad \boldsymbol{T} = 2\eta_p \begin{pmatrix} 1 & -4\lambda U y \\ 0 & 1 \end{pmatrix} \tag{6.7}$$

where $\eta_p = 1$ and $U = 0.1$. Tables 6.7 and 6.8 display the error computations between the analytical and numerical results, conducted for various relaxation parameter λ values, using both uniform and perturbed meshes obtained via the decoupled and monolithic formulation, respectively. In contrast to the EVSS formulation, with the Tensor Stokes formulation, we achieved successful results even at very high relaxation parameter values, up to $\lambda = 30$, using the decoupled formulation. However, with the monolithic formulation, we encountered limitations in obtaining promising results for relaxation parameter values beyond $\lambda = 0.1$.

	Tensor Stokes decoupled formulation			
λ	$\|\|\mathbf{u}_{analytic} - \mathbf{u}\|\|$		$\|\|\boldsymbol{T}_{analytic} - \boldsymbol{T}\|\|$	
	Uniform	Perturbed	Uniform	Perturbed
0.0	1.089283e-15	4.551555e-16	9.762144e-13	1.612860e-12
0.1	1.373892e-15	3.670640e-16	9.915842e-12	8.175144e-12
1.0	4.558579e-15	4.664982e-16	2.840642e-11	2.409553e-11
5.0	1.058596e-14	1.433698e-15	9.001235e-11	5.885064e-11
10.0	2.779410e-14	2.998034e-15	2.756288e-10	1.084233e-10
20.0	1.012888e-13	1.126973e-14	1.385069e-09	4.554372e-10
30.0	2.205335e-13	2.735274e-14	4.793212e-09	1.806251e-09

Table 6.7: **Tensor Stokes decoupled Oldroyd-B fluid:** Error computations for the Oldroyd-B flow for increasing values of the relaxation parameter λ obtained via Tensor Stokes decoupled formulation.

	Tensor Stokes Monolithic formulation			
λ	$\|\|\mathbf{u}_{analytic} - \mathbf{u}\|\|$		$\|\|\boldsymbol{T}_{analytic} - \boldsymbol{T}\|\|$	
	Uniform	Perturbed	Uniform	Perturbed
0.001	1.094561e-14	6.607592e-15	2.679392e-11	1.863350e-11
0.010	6.122942e-15	4.607492e-15	1.372338e-11	2.039642e-11
0.100	2.371823e-14	2.248952e-14	1.440208e-09	1.774980e-09

Table 6.8: **Tensor Stokes monolithic Oldroyd-B fluid:** Error computations for the Oldroyd-B flow for increasing values of the relaxation parameter λ obtained via Tensor Stokes monolithic formulation.

Fig. 6.14 and 6.15 illustrates the components of the tensor $\boldsymbol{T}$ on both the uniform and perturbed meshes, obtained through the decoupled and monolithic formulations, respectively. It can be observed that the components of tensor $\boldsymbol{T}$ are in good agreement with the analytic $\boldsymbol{T}$ given in equation (6.7).

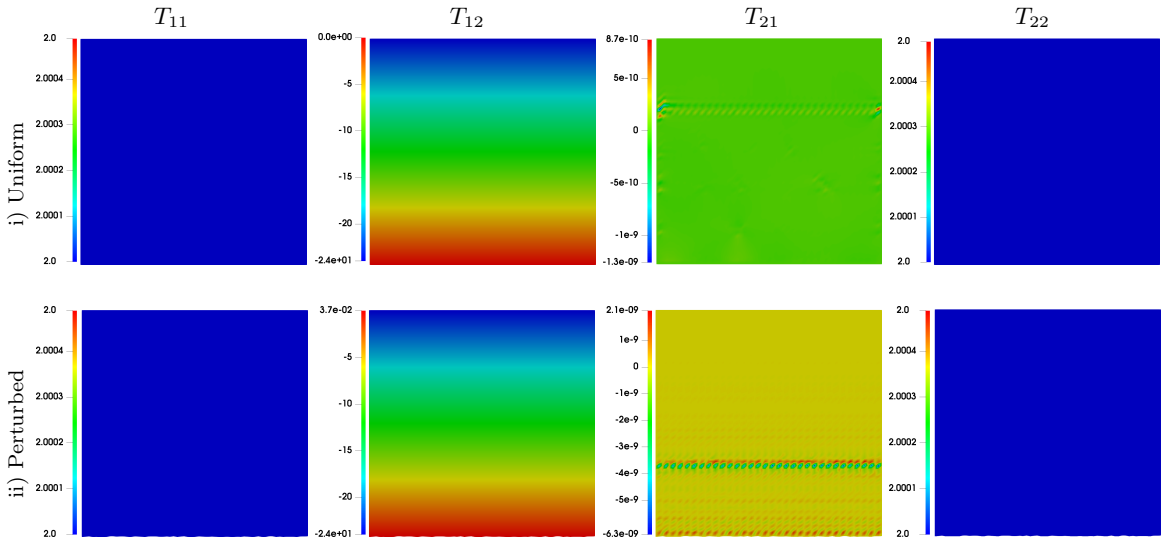


Fig. 6.14: **Decoupled approach for Oldroyd-B fluid:** Components of the stress tensor $\boldsymbol{T}$ obtained via decoupled formulation at $\lambda = 30$ at mesh refinement level 6.

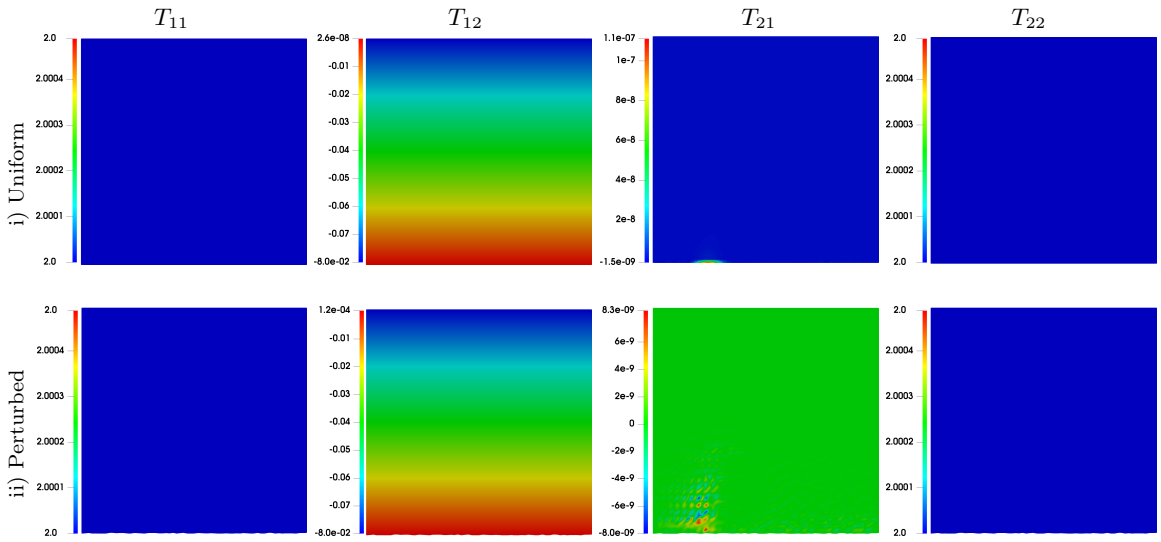


Fig. 6.15: **Monolithic approach for Oldroyd-B fluid:** Components of the stress tensor $\boldsymbol{T}$ obtained via monolithic formulation at $\lambda = 0.1$ at mesh refinement level 6.

In this chapter, the Newtonian results for the flow around the cylinder configuration and stick-slip problem is presented. Following the successful validation against benchmarks, viscoelastic simulations for the simplest Oldroyd-B model were conducted using all three formulations. These simulations computed the error between analytic and numerical results for increasing values of the relaxation parameter λ. With satisfactory outcomes obtained for the simplest Oldroyd-B model, the next chapter is dedicated to a more complex geometry i.e., four-to-one contraction flow.

7

Application to the contraction flow

In this chapter of the thesis, let us consider a close enough configuration for the engineering and industrial applications, the 4:1 contraction configuration, to analyze the viscoelastic phenomena. Apart from its importance in engineering and industrial applications, contraction flows are also utilized in the realms of applied mathematics and numerical simulation. The contraction flow offers valuable insights for developing effective numerical methods and practical algorithms to tackle the complex flow properties. The unique aspect of this configuration lies in its reentrant corner, which results in the formation of recirculating flows. Keunings [34] investigated the impact of corners on contraction flows. In order to simplify the problem and concentrate on the flow characteristics near the channel's wall, we opt to eliminate the challenges posed by sharp reentrant corners by replacing them with smoothed ones. The rounded geometry is generated by Brown et al. [15] with a curve segment at contraction to reduce the severe stress. Moreover, taking advantage of the symmetry in the 4:1 contraction, only half of the domain is utilized for the computations. This reduces the number of elements to half for the numerical computations and thus provide somewhat numerical ease. The computational domain for the 4:1 contraction along with the coarse mesh is shown in Fig. 7.1 where the in- and outflow edges are of height $(y_{in} \in (0, +4)\,, y_{out} \in (0, +1))$, respectively. The boundary, $\partial\Omega = \Gamma_{\text{in}} \cup \Gamma_{\text{no-slip}} \cup \Gamma_{\text{out}} \cup \Gamma_{\text{sym}}$, is composed of four sections, inflow entrance section denoted by Γ_{in}, upper channel wall $\Gamma_{\text{no-slip}}$, lower channel wall Γ_{sym} and outflow exit section Γ_{out}.

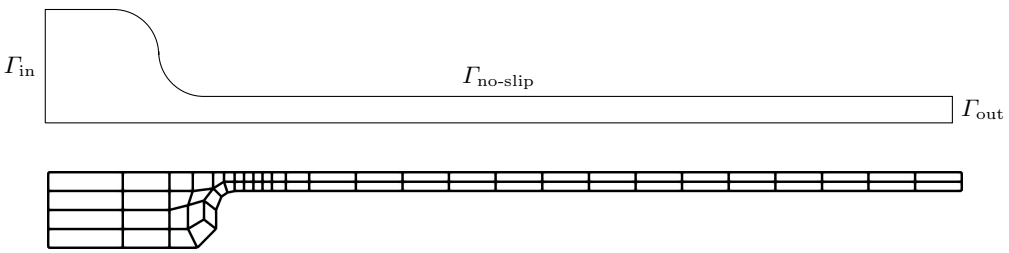


Fig. 7.1: Configuration for 4:1 contraction flow

The length of the channel is set equal to L. Details regarding the computational meshes utilized are provided in Table 7.1.

Level l	Nodes	Edges	Elements	Degree of freedom
0	93	154	62	3400
1	309	556	248	12622
2	1113	2104	992	48550
3	4209	8176	3968	190342
4	16353	32224	15872	753670

Table 7.1: Mesh details for the rounded contraction flow

The nonuniform geometry of the curved contraction depicted in Fig. 7.1 presents mesh discretization challenges. In addition to managing the 4:1 aspect ratio of the contraction, mesh generation has to handle the curved corners. The entrance and exit sections of the contraction are discretized using structured quadrilaterals, while the transition patch employs a combination of both structured and unstructured quadrilaterals. To thoroughly examine the flow characteristics, we let the flow develop freely after the entrance and use a longer exit zone for the outflow to prevent any non-physical behavior.
The detailed study on the curved contraction flow begins by considering the simplest viscoelastic Oldroyd-B fluid using the standard formulation in a decoupled manner. As already discussed, the standard viscoelastic formulation has the limitation to be solved in a decoupled manner in the absence of solvent viscosity. Therefore, the formulation is regularized in a suitable manner such that the added stabilization terms on both sides of the momentum equation scaled by β helps to obtain the convergence without affecting the end solution (5.3, 5.4). Concerning boundary conditions, a fully developed velocity profile for the Oldroyd-B model is imposed on the inflow Γ_{in} and outflow Γ_{out} edges of the computational domain, together with no-slip $\Gamma_{\text{no-slip}}$ on the upper and the symmetry Γ_{sym} condition on the lower wall. Stress profiles are assigned specifically at the inflow edge Γ_{in}, while do-nothing boundary conditions for the stress tensor are applied to all other boundaries. The following parabolic velocity and stress functions are prescribed at the inflow of the contraction,

$$\mathbf{u} = \left(U/16\left(16 - y^2\right), 0\right) \qquad \boldsymbol{S} = \begin{pmatrix} \frac{2}{64}\lambda\eta_p U^2 y^2 & -\frac{1}{8}\eta_p U y \\ -\frac{1}{8}\eta_p U y & 0 \end{pmatrix} \tag{7.1}$$

where $\eta_p = 1$ and $U = 0.1$. The outflow velocity profile is given as,

$$\mathbf{u} = \left(4U\left(1 - y^2\right), 0\right). \tag{7.2}$$

where the velocity of $U = 0.1$ gives the mean velocity $U_{\text{mean}} = 0.27$ at the downstream of the channel having height $h = 1$. The Weissenberg number for the resulting flow is computed as $\text{We} = \lambda \frac{U_{\text{mean}}}{1} = 0.27\lambda$. The numerical computations are performed using UMFPACK solver for the Stokes problem as well as constitutive equation. The tabular and the graphical representation of the relationship between the relaxation λ and stabilization parameter β can be seen in Fig. 7.2. The Table shows the dependence of the relaxation time λ on the stabilization parameter β and the number of the iterations required for the convergence at mesh refinement level 3. It can be observed that relaxation parameter λ and diffusion stabilization parameter β are directly proportional to each other. That means with increasing relaxation parameter λ value, one has to add more diffusion in order to obtain the convergence. This technique can be termed as a hit and trial method to find the stabilization parameter β to obtain the convergence. The considered method would lead to highly cumbersome numerical computations and sensitive formulation for high values of the relaxation parameter λ values as well as for the complex viscoelastic models.

λ	β	No. of iterations
0.0	0.8	16
0.1	0.8	27
0.5	0.8	203
1.0	0.9	603
1.1	0.9	700
1.2	1.1	968
1.3	1.2	1200
1.4	1.3	1461
1.5	1.6	1995

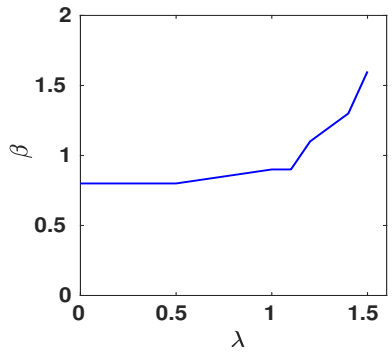


Fig. 7.2: Dependence of relaxation parameter λ on the stabilization parameter β

The next step is to formulate an algorithm that would choose the diffusion stabilization parameter β on it's own. At the first step, let us begin with the idea proposed by Westervoss [66] and compute the tensor $\boldsymbol{T}$ by the post processing of the results obtained by the three field formulation. The element wise constant $\boldsymbol{T} \in \mathbb{R}^{2\times 2}$ is obtained by the post processing of the velocity $\mathbf{u}$ and stress tensor $\boldsymbol{S}$ computed from the decoupled differential viscoelastic formulation in $\mathbf{u}, p, \boldsymbol{S}$. The tensor $\boldsymbol{T}$ is calculated by the following relation,

$$\boldsymbol{S} = \boldsymbol{T} \cdot \mathbf{D}(\mathbf{u}) \tag{7.3}$$

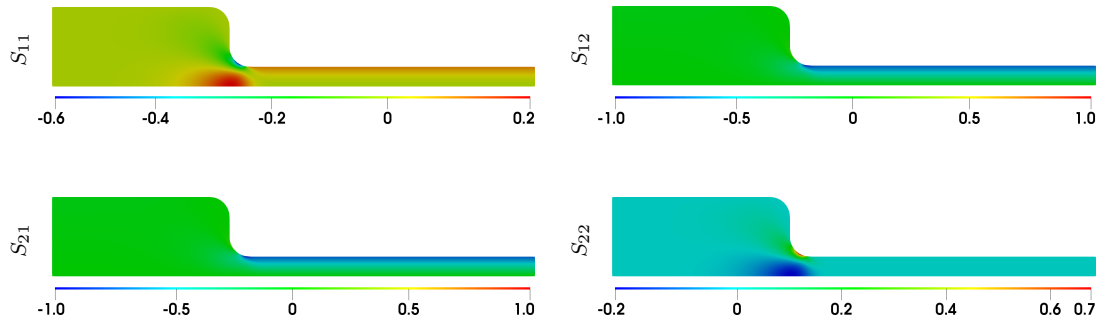


Fig. 7.3: **Extra-stress tensor components for contraction flows:** Components of tensor $\boldsymbol{S}$ using Q_3 element, for Oldroyd-B model at relaxation parameter $\lambda = 0.1$ at mesh refinement level 3.

where $\boldsymbol{D}$ and $\boldsymbol{S} \in \mathbb{R}^{2\times 2}$. The numerical simulation are performed at relaxation parameter $\lambda = 0.1$ to obtain the velocity, pressure and stress results. The tensor $\boldsymbol{T}$ is computed by the postprocessing of these results. The graphical representation of the stress tensor $\boldsymbol{S}$ and element wise constant $\boldsymbol{T}$ are plotted in Fig. 7.3 and 7.4, respectively.

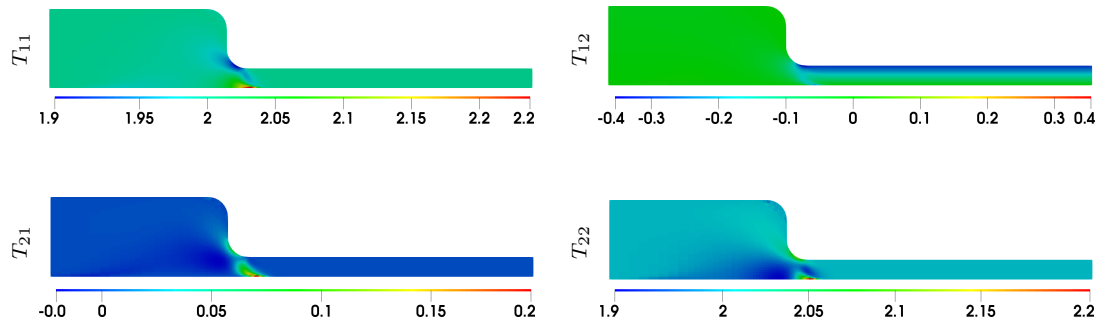


Fig. 7.4: **Tensor Stokes components for contraction flows:** Components of tensor $\boldsymbol{T}$ for Oldroyd-B model at relaxation parameter $\lambda = 0.1$ at mesh refinement level 3.

In order to develop a robust decoupled algorithm a modification in the parameter β is employed. Instead of employing a random constant diffusion stabilization parameter β in the decoupled formulation the components of the tensor $\boldsymbol{T}$ are employed to compute the optimal value of the parameter β. Let us consider the four field formulation such that the tensor $\boldsymbol{T}$ is not computed at the post processing step but it is computed in each iterative step where its components are utilized and stabilization parameter in diffusive part of the Stokes problem is updated accordingly.

Let $(\boldsymbol{u}^n, p^n, \boldsymbol{S}^n, \boldsymbol{T}^n)$ be the known approximation of $(\mathbf{u}, p, \boldsymbol{S}, \boldsymbol{T})$ after n steps. The step $(n+1)$ of the algorithm consists of first computing $(\mathbf{u}^{n+1}, p^{n+1})$ by solving the Stokes problem using the previous stress $\boldsymbol{S}^n$ in the right-hand side of the linear system, and then computing $\boldsymbol{S}^{n+1}$ by using its constitutive equation, with the new velocity $\mathbf{u}^{n+1}$ in the right-hand side. In the third step, the tensor $\boldsymbol{T}^{n+1}$ is computed from the velocity and stress. The constant value β_c^{n+1} is computed from the components of the tensor $\boldsymbol{T}$. The iteration $(n+1)$ consists of first computing $(\mathbf{u}^{n+1}, p^{n+1})$,

$$\begin{cases} \nabla p^{n+1} - \nabla \cdot \beta_c^n \mathbf{D}\left(\mathbf{u}^{n+1}\right) = \nabla \cdot \boldsymbol{S}^n - \nabla \cdot \beta_c^n \mathbf{D}\left(\mathbf{u}^n\right) \\ \nabla \cdot \mathbf{u}^{n+1} = 0 \end{cases} \tag{7.4}$$

In the second step we compute $\boldsymbol{S}^{n+1}$,

$$\boldsymbol{S}^{n+1} + \lambda \overset{\nabla}{\boldsymbol{S}}^{n+1} = 2\eta_p \mathbf{D}\left(\mathrm{u}^{n+1}\right) \tag{7.5}$$

and in the third step the tensor $\boldsymbol{T} \in \mathbb{R}^{2\times 2}$ at the center of the cell is computed by the following relation,

$$\boldsymbol{T}^{n+1} = \boldsymbol{S}^{n+1} \cdot \mathbf{D}^{-1}\left(\mathbf{u}^{n+1}\right) \tag{7.6}$$

where $\mathbf{D}$ and $\boldsymbol{S} \in \mathbb{R}^{2\times 2}$. The constant value β_c is updated in every iteration with the help of the tensor $\boldsymbol{T}$ that is calculated at the center of the cell. There are vast number of possibilities that can be employed to the components of the tensor $\boldsymbol{T}$ to select the constant value β_c. Below mentioned are the few examples that utilize the components of the tensor $\boldsymbol{T}$ to compute β_c.

Parameter β_c	Components of the tensor $\boldsymbol{T}$
Trace	$T_{11} + T_{22}$
Mean	$\frac{1}{2}\left(T_{11} + T_{22}\right)$
Average	$\frac{1}{4}\left(T_{11} + T_{12} + T_{21} + T_{22}\right)$
$\sqrt{\text{Average}}$	$\sqrt{\frac{1}{4}\left(T_{11} + T_{12} + T_{21} + T_{22}\right)}$
Norm	$\sqrt{\left({T_{11}}^2 + {T_{12}}^2 + {T_{21}}^2 + {T_{22}}^2\right)}$

Table 7.2: Few examples for utilization of the components of tensor $\boldsymbol{T}$ to compute β_c.

The numerical results are computed using the formulas described in Table 7.2. These are the few examples and of course one could play around and produce further versions of the stabilization parameter β. The advantage of such algorithm is the possibility to

be able to select the diffusion stabilization parameter on it's own rather than a hit and trial constant β. Table 7.3 shows the number of iterations required for convergence for increasing relaxation parameter λ values for the constant diffusion stabilization parameter as well as β_c obtained using the novel algorithm.

The idea proposed by Westervoss [66] helped in the development of the modified decoupled algorithm. The recent update to the diffusion constant β_c within the algorithm, irrespective of the relaxation parameter λ, has sparked the hope for an efficient algorithm capable of updating the scaling factor alongside the diffusive term. However, it can be noticed in Table 7.3. the number of iterations required for convergence are higher for the adaptive parameter β_c as compared to the constant β. This shows that we have not achieved an ideal algorithm because on one hand our problem size is increased to four field formulation and on the other hand it does not give promising results in terms of the number of iterations.

Let's move forward with the aim to work on an algorithm that can be treated in a decoupled way without the need for additional stabilization. While dealing with the realistic flow problems that is simulating pure polymer melts the aim is to recover the diffusive operator in the momentum equation without artificial stabilization but the application of the change of variables.

The EVSS and Tensor Stokes formulation introduce the splitting of the stress tensor such that change of variables is performed in the original equations and velocity coupling is reintroduced back into the momentum equation. The re-introduction of the velocity coupling omits the need of the artificial diffusion in the momentum equation.

The numerical results for the EVSS and Tensor Stokes formulation are computed for the contraction flow. The potential benefits of these two formulation includes that one could solve for the three field formulation with zero solvent viscosity in a decoupled way without any added stabilization, and offers the possibility for the application of the multigrid solver in a monolithic formulation. In the Tensor Diffusion approach the viscosity constant that scales the diffusion tensor is replaced by the viscosity tensor that is computed and updated within the algorithm. We can say that the Tensor Stokes three field formulation have the natural type of the diffusive operator in the momentum equation for the realistic flows. The introduction of a numerical algorithm with the natural type of the diffusive operator that recovers the velocity coupling in the momentum equation is novel, and it has the potential to be explored using both fully monolithic and decoupled methods for various viscoelastic fluid models. Both the monolithic and decoupled approaches are used in the following sections to obtain results for viscoelastic problems.

β_c	λ	No. of iterations
1	0	7
	0.01	7
	0.1	65
Mean	0	15
	0.01	16
	0.1	137
Average	0	9
	0.01	14
	0.1	–
$\sqrt{\text{Average}}$	0	7
	0.01	10
	0.1	65
Norm	0	22
	0.01	22
	0.1	193
Trace	0	30
	0.01	32
	0.1	270

Table 7.3: **Oldroyd-B contraction flow:** Number of iterations required for convergence at mesh refinement level 3 with constant $\beta = 1$ and β_c obtained by utilizing the components of the tensor $\boldsymbol{T}$.

7.1 Monolithic approach: EVSS formulation

This section presents the monolithic results obtained using the steady-state EVSS approach for the three viscoelastic fluid models. The convergence criterion was established such that the norm of the defect vectors is less than a specified tolerance of 10^{-6}.

7.1.1 Oldroyd-B model

Among the viscoelastic models, Oldroyd-B model is the simplest and therefore often selected as the first model for numerical simulations because of the availability of the analytic functions for the velocity and stress. The numerical results for the Oldroyd-B fluid, with increasing values of lambda λ, are examined both with and without the EO

stabilization parameter $\gamma_{\boldsymbol{E}}$ in the stress equation. The EVSS formulation for the Oldroyd-B model is solved subject to the following boundary conditions,

- The parabolic velocity profiles

$$\mathbf{u} = \left(U/16\left(16 - y^2\right), 0\right) \qquad \text{and} \qquad \mathbf{u} = \left(4U\left(1 - y^2\right), 0\right). \tag{7.7}$$

  are prescribed at the inflow and outflow, respectively.

- On the upper wall and lower wall set no slip and symmetry boundary condition, respectively.

- EVSS stress profile based on the inflow velocity is prescribed only at the inflow

$$\boldsymbol{E} = \begin{pmatrix} \frac{1}{32}\lambda\eta_p U^2 y^2 & 0 \\ 0 & 0 \end{pmatrix} \tag{7.8}$$

where $\eta_p = 1$ and $U = 0.4$. The Newtonian flow is recovered at $\lambda = 0$ while the viscoelastic results are obtained for $\lambda > 0$. The limits of the relaxation parameter λ are recorded in Table 7.4.
The table shows the number of iterations required for convergence for increasing values of the relaxation parameter λ. As the viscoelastic effects are intensified by increasing the relaxation parameter λ the difficulty level to solve the problem increases. The addition of the EO stabilization not only helps to achieve the results at higher relaxation parameter λ but also requires less number of iterations to achieve convergence at lower λ values. The addition of EO stabilization to the stress component helps to achieve the results at $\lambda = 2.5$ where no convergence could be achieved at $\lambda = 2$ without EO stabilization.

λ	$\gamma_{\boldsymbol{E}} = 0.0$	$\gamma_{\boldsymbol{E}} = 0.01$
0.0	2	2
0.1	3	2
0.5	17	4
1.0	50	12
1.5	143	19
2.0	-	36
2.5	–	120

Table 7.4: **Limits of λ for Oldroyd-B model:** Number of iterations required for convergence of the Oldroyd-B model for increasing values of the relaxation parameter λ, and EO stabilization parameter $\gamma_{\boldsymbol{E}} = 0.0$ and $\gamma_{\boldsymbol{E}} = 0.01$.

The convergence plots in Fig. 7.5 demonstrate that applying the EO stabilization significantly accelerates convergence. In contrast, its absence results in a substantially higher

number of iterations required to meet the tolerance criteria. Specifically, the stabilization method reduces the computational effort needed to achieve the desired results.

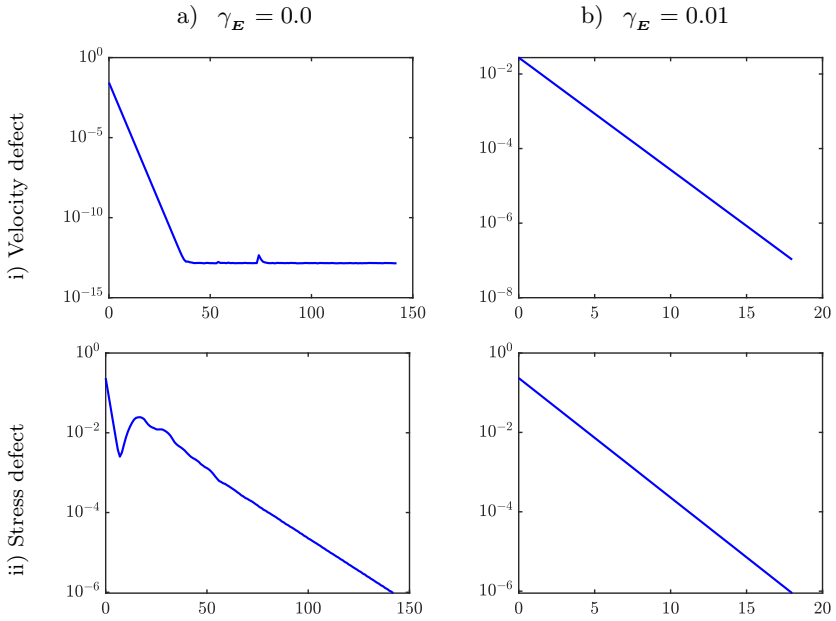


Fig. 7.5: **Convergence plots Oldroyd-B model:** Number of iterations required for convergence of the Oldroyd-B model with relaxation parameter value λ=1.5, and EO stabilization parameter $\gamma_{\boldsymbol{E}} = 0.0$ and $\gamma_{\boldsymbol{E}} = 0.01$.

The components of the stress tensor $\boldsymbol{E}$ are displayed in Fig. 7.6. The addition of EO stabilization to the stress equation significantly improves the speed of convergence. This means that the solution reaches the desired accuracy faster when EO stabilization is applied. Importantly, this improvement in convergence does not compromise the quality of the solution itself atleast on the visual basis. The reason behind this enhanced convergence and good quality of solution could be the use of a higher order finite element, specifically Q_3, for the stress equation. Higher order finite elements like Q_3 are known for their ability to capture more complex variations in the solution, providing greater accuracy. Thus, while EO stabilization accelerates the convergence, the Q_3 elements ensure that the solution remains reliable. This combination allows for both efficient computation and high quality results.

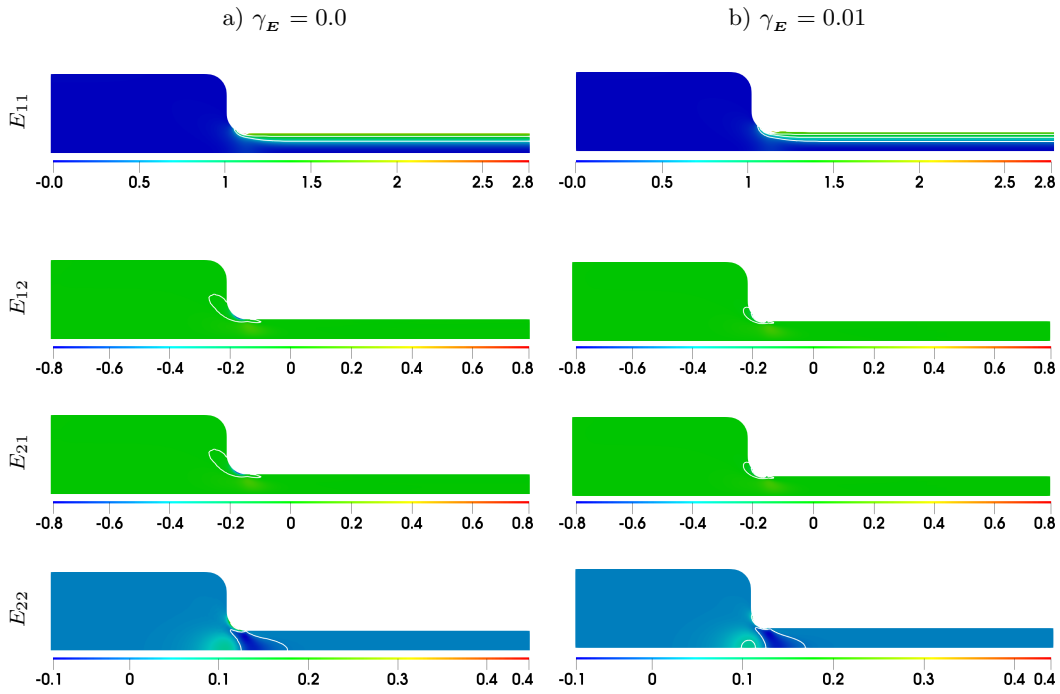


Fig. 7.6: **Oldroyd-B contraction flow:** Components of tensor $\boldsymbol{E}$ using Q_3 for Oldroyd-B model at relaxation parameter $\lambda = 1.5$, and EO stabilization parameter $\gamma_{\boldsymbol{E}} = 0.0$ and $\gamma_{\boldsymbol{E}} = 0.01$ at mesh refinement level 3.

As already mentioned, Oldroyd-B is the simplest viscoelastic model such that it is unable to capture any realistic or meaningful viscoelastic material behaviour. For example, this can be realised based on the corresponding fully developed parabolic velocity profile from the velocity function in equation (7.7), which is independent of the relaxation parameter λ. Consequently, no modeling of any nonlinear effects such as shear thinning, which occurs for increasing relaxation parameter, is possible. Thus, results for the Giesekus and PTT exponential model are computed that indeed depicts the viscoelastic effects such that the flow profiles deviate from the parabolic shape.

7.1.2 Giesekus model

This section explores the monolithic EVSS Giesekus fluid formulation for various relaxation parameter values λ. The results are derived both with and without the inclusion of the EO stabilization parameter $\gamma_{\boldsymbol{E}}$ to the stress component of the constitutive equation. The Giesekus model is derived by adding an extra quadratic stress term to the Oldroyd-B model. By defining the nonlinear material parameter α within the range of

$(0, 1]$, the model can effectively anticipate a shear thinning phenomenon. When considering parabolic profiles as the inflow function to compute fully developed viscoelastic flow profiles in simulating viscoelastic Poiseuille flow using the Giesekus model, inappropriate flow phenomena is observed. It is clear that the selected parabolic velocity profile isn't appropriate, as the velocity field displays a distinct shape away from the inflow and outflow boundaries. This contradicts the expected numerical results for Poiseuille flow as shown by the study of Westervoss [66]. Therefore, the fully developed flow profiles resulting in case of the Giesekus model are evaluated with respect to the included material and model parameter and pressure drop with the help of the following semi-analytic expression of the velocity profile in the non-solvent case given in [21]. The EVSS formulation for the Giesekus model is solved subject to the following boundary conditions:

- On the left and right edge the semi-analytic 1-D velocity profiles are prescribed. The velocity profile for the zero solvent viscosity η_s=0 is an analytic function [21, 66] that depends upon the material parameters and pressure drop.

$$u_x(y) = \frac{B}{\lambda}\Big[\left(1-2\beta^2\right) log\frac{\beta+\sqrt{1-\phi^2\frac{y^2}{B^2}}}{\beta+\sqrt{1-\phi^2}} + \beta\left(\sqrt{1-\phi^2\frac{y^2}{B^2}}-\sqrt{1-\phi^2}\right) + \beta\left(1-\beta^2\right)\left(\frac{1}{\beta+\sqrt{1-\phi^2\frac{y^2}{B^2}}}-\frac{1}{\left(\beta+\sqrt{1-\phi^2}\right)}\right)\Big]/\phi \tag{7.9}$$

  $u_y = 0$. In the above expression $\beta = 2\alpha - 1$ and $\phi = 2\alpha\lambda P_x$ where $\alpha = 0.1$.

- On the upper and lower wall set no slip boundary condition and symmetry boundary conditions, respectively.

- On the left edge the semi-analytic stress profile $\boldsymbol{E}$ is prescribed

$$\boldsymbol{E} = \begin{pmatrix} \frac{(1-\alpha)\left(1\mp\sqrt{1-4\alpha^2\frac{\lambda^2P_x^2y^2}{B^2}}\right)+2\alpha^2\frac{\lambda^2P_x^2y^2}{B^2}}{\alpha\left(2\alpha-1\pm\sqrt{1-4\alpha^2\frac{\lambda^2P_x^2y^2}{B^2}}\right)} & \frac{\lambda P_x y}{B}-\eta_p\frac{\partial u}{\partial y} \\ \frac{\lambda P_x y}{B}-\eta_p\frac{\partial u}{\partial y} & \frac{-1\pm\sqrt{1-4\alpha^2\frac{\lambda^2P_x^2y^2}{B^2}}}{2\alpha} \end{pmatrix}.$$

where,

$$\frac{\partial u}{\partial y} = \frac{2\alpha\lambda P_x y}{B}\frac{1\pm(2\alpha-1)\sqrt{1-\frac{4\alpha^2\lambda^2P_x^2y^2}{B^2}}}{\left(2\alpha-1\pm\sqrt{1-\frac{4\alpha^2\lambda^2P_x^2y^2}{B^2}}\right)^2}$$

Ferras et al. [21] shows that the lower branch solution exhibits physically unrealistic solutions and therefore only the upper branch solution can be considered for the numerical

simulations. Table 7.5 shows the number of iterations required for convergence for increasing values of the relaxation parameter λ. The addition of EO stabilization parameter $\gamma_{\boldsymbol{E}}$ helps to achieve results at comparatively higher values of the relaxation parameter λ in less number of iterations as compared to the absence of the EO stabilization parameter value. Fig. 7.7 shows the convergence plots at $\lambda = 2$, it can be observed that the addition of EO stabilization parameter reduces the number of iterations almost near to half.

λ	$\gamma_{\boldsymbol{E}} = 0.0$	$\gamma_{\boldsymbol{E}} = 0.01$
0.1	4	4
0.5	5	5
1.0	21	20
2.0	122	58
2.5	483	67
3.0	–	87
3.5	–	97

Table 7.5: **Limits of λ for Giesekus model:** Number of iterations required for convergence for Giesekus model at $\alpha = 0.1$, for increasing values of parameter λ.

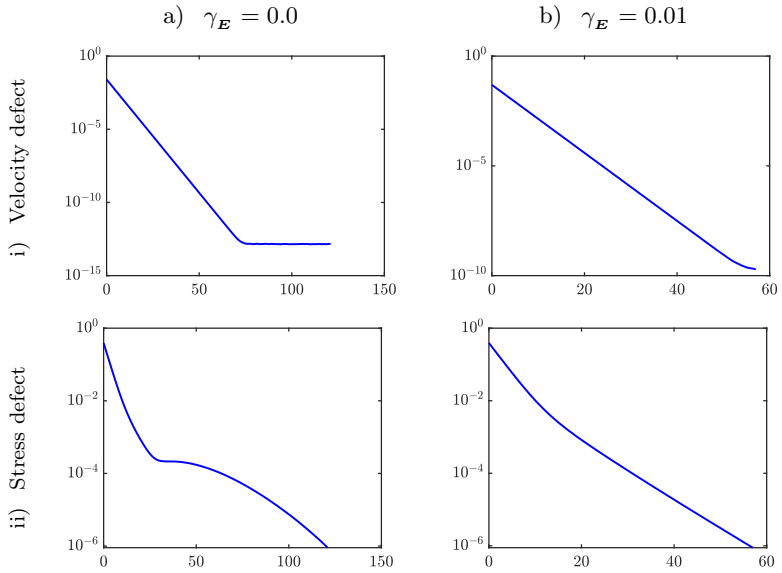


Fig. 7.7: **Convergence plots for Giesekus model:** Number of iterations required for convergence for Giesekus model at λ=2, and EO parameter $\gamma_{\boldsymbol{E}} = 0.0$ and $\gamma_{\boldsymbol{E}} = 0.01$.

This shows that the addition of EO stabilization parameter not only helps to obtain results at higher values of the relaxation parameter λ but also aids in the faster convergence. Fig. 7.8 is plotted to graphically visualize the solution with and without the presence of EO stabilization parameter at relaxation parameter value $\lambda = 2.5$.

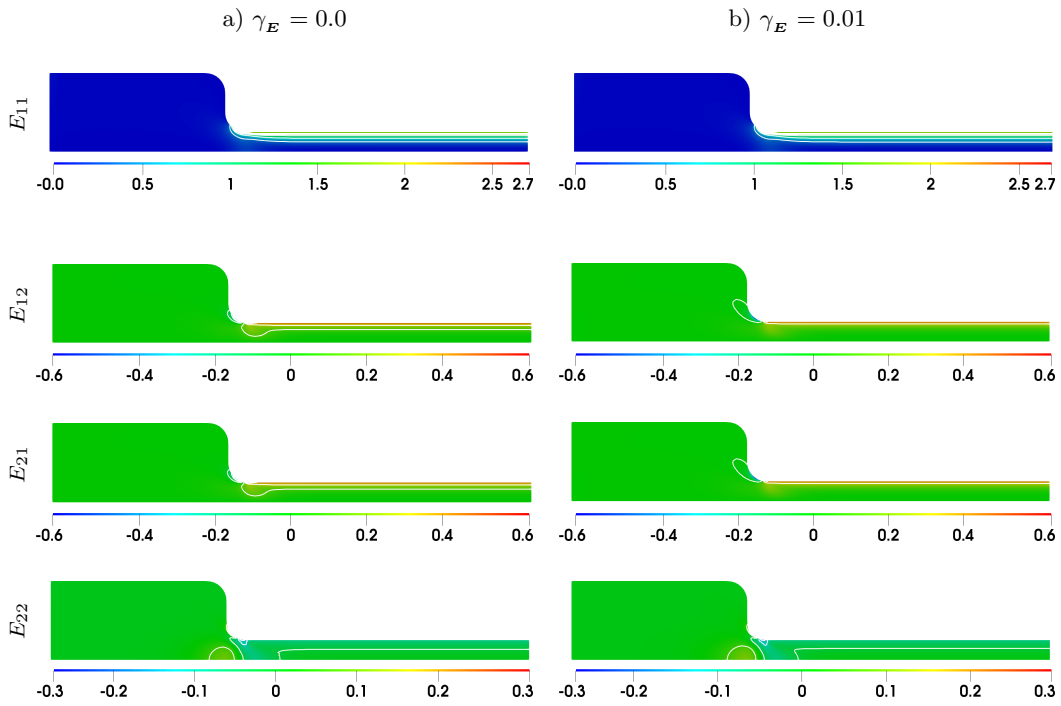


Fig. 7.8: **Giesekus contraction flow:** Components of tensor $\boldsymbol{E}$ for Giesekus model at $\lambda = 2.5$, and EO parameter $\gamma_{\boldsymbol{E}} = 0.0$ and $\gamma_{\boldsymbol{E}} = 0.01$ at mesh refinement level 3.

The similar numerical solution technique needs to be applied when considering the PTT exponential model. This model offers comparable numerical solutions to those of the Giesekus model. However, for thoroughness, the next section briefly examines the PTT exponential model.

7.1.3 PTT exponential model

In this section, we are examining how the flow profiles in the PTT exponential model change based on the included material parameter included and how it affects the nonlinearity of the solution. The numerical results are prepared both with and without incorporating the EO stabilization parameter $\gamma_{\boldsymbol{E}}$ in the stress equation. Just like the Giesekus

model, the PTT exponential model can predict a shear thinning effect by setting the nonlinear material parameter $\kappa = 0.1$. The EVSS formulation for the PTT-exponential model is solved subject to the following boundary conditions:

- On the left and right edge the semi-analytic 1-D velocity profiles are prescribed. The velocity profile for the zero solvent viscosity η_s=0 is an analytic function [21] that depends upon the material parameters and the pressure drop,

$$u_x = \frac{\eta_p}{4\kappa\lambda^2\frac{\partial p}{\partial x}}\left[exp\left(2\kappa\left(\frac{\lambda}{\eta_p}\frac{\partial p}{\partial x}\left(y-\frac{b+a}{2}\right)\right)^2\right) - exp\left(2\kappa\left(\frac{\lambda}{\eta_p}\frac{\partial p}{\partial x}\left(\frac{b-a}{2}\right)\right)^2\right)\right] \tag{7.10}$$

  $u_y = 0$.
- On the upper wall set no slip boundary condition.
- On the lower wall set symmetry boundary condition.
- On the left edge the semi-analytic stress profile $\boldsymbol{E}$ [21] is prescribed,

$$\boldsymbol{E} = \begin{pmatrix} \frac{2\lambda}{\eta_p}\left(\frac{\partial p}{\partial x}y\right)^2 & \frac{\partial p}{\partial x}y\left(1-exp\left(2\kappa\left(\frac{\lambda}{\eta_p}\frac{\partial p}{\partial x}y\right)^2\right)\right) \\ \frac{\partial p}{\partial x}y\left(1-exp\left(2\kappa\left(\frac{\lambda}{\eta_p}\frac{\partial p}{\partial x}y\right)^2\right)\right) & 0 \end{pmatrix}$$

It can be seen that the velocity function is characterized by a $y-$dependent term in the $x-$direction, while the stress tensor components vary solely with respect to the channel height. The application of these properties to the steady-state PTT exponential and Giesekus model, yields a nonlinear system of ordinary differential equations (ODEs) in one dimension. The pressure drop $\frac{\partial p}{\partial x}$ in the provided function is determined by solving a one-dimensional nonlinear set of equations, ensuring that the velocity profile achieves the desired flow rate. After finding the pressure drop, we use it to set up the necessary boundary conditions for velocity and stress functions in our 2-D code. The similar technique is utilized to find the pressure drop for the Giesekus model.
Table 7.6 shows the number of iterations required for convergence for different values of the relaxation parameter λ. Fig. 7.9 shows the convergence plots at relaxation parameter $\lambda = 2$ and Fig. 7.10 is plotted to graphically visualize the solution with and without the presence of EO stabilization parameter at relaxation parameter $\lambda = 2$. The numerical findings for the three viscoelastic models exhibit similar trends regarding the application of EO stabilization and its impact on the convergence behavior.

λ	$\gamma_{\boldsymbol{E}} = 0.0$	$\gamma_{\boldsymbol{E}} = 0.01$
0.1	5	5
0.5	8	7
1.0	31	23
2.0	319	53
2.5	-	52
3.0	–	55
3.5	–	70
4.0	–	271
5.0	–	211
6.0	–	220
7.0	–	227

Table 7.6: **Limits of λ for PTT exponential model:** Number of iterations required for convergence of the PTT exponential model at $\kappa = 0.1$, for increasing values of λ.

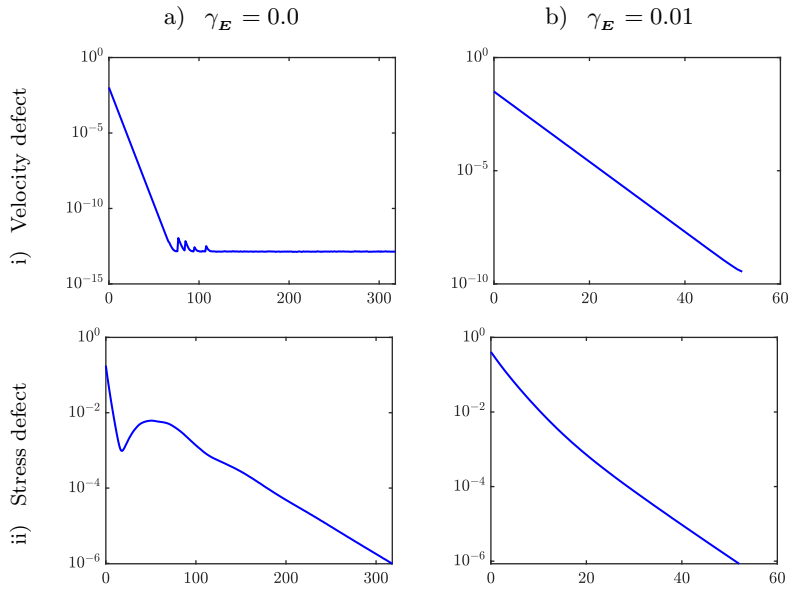


Fig. 7.9: **Convergence plots for PTT exponential model:** Number of iterations required for convergence of the PTT exponential model at λ=2.

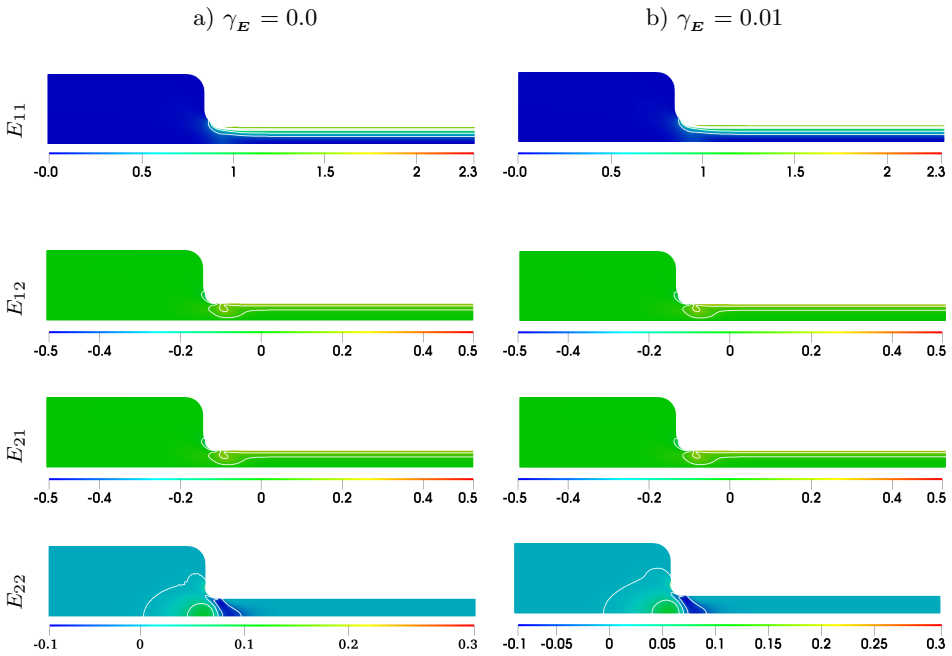


Fig. 7.10: **PTT exponential contraction flow:** Components of tensor $\boldsymbol{E}$ for PTT exponential model at $\lambda = 2$, and EO stabilization parameter $\gamma_{\boldsymbol{E}} = 0.0$ and $\gamma_{\boldsymbol{E}} = 0.01$ at mesh refinement level 3.

In the following section, we will revisit the viscoelastic models and study the numerical results for the novel Tensor Stokes formulation.

7.2 Monolithic approach: Tensor Stokes formulation

In this section, we showcase the monolithic results achieved through the steady-state Tensor Stokes approach for the three viscoelastic fluid models. The convergence criterion was set to ensure that the norm of the defect vectors is below a specified tolerance of 10^{-6}. Let us begin our numerical investigation with the simplest Oldroyd-B model.

7.2.1 Oldroyd-B model

Similar to the EVSS formulation, Oldroyd-B model is the first choice to test for the Tensor Stokes formulation due to its simplicity. We examine the numerical results of the Oldroyd-B fluid as the relaxation parameter λ values increase, both with and without the

addition of EO stabilization parameter in the constitutive equation of the Tensor Stokes formulation. The Tensor-Stokes formulation for the Oldroyd-B model is solved subject to the following boundary conditions.

- The parabolic velocity profiles

$$\mathbf{u} = \left(U/16\left(16 - y^2\right), 0\right) \qquad \text{and} \qquad \mathbf{u} = \left(4U\left(1 - y^2\right), 0\right). \tag{7.11}$$

  are prescribed at the inflow and outflow, respectively.
- On the upper wall and lower wall set no slip boundary condition and symmetry boundary condition, respectively.
- Analytic stress profile $\boldsymbol{T}$ based on the inflow velocity is prescribed only at the inflow

$$\boldsymbol{T} = \eta_p \begin{pmatrix} 2 & -\frac{1}{2}\lambda U y \\ 0 & 2 \end{pmatrix} \tag{7.12}$$

Table 7.7 reflects the importance of EO stabilization parameter $\gamma_{\boldsymbol{T}}$ in the constitutive equation. The addition of EO stabilization in the constitutive equation helps to obtain results at higher lambda values as compared to no EO stabilization. The convergence plots also highlight the significance of incorporating EO stabilization. Specifically, for a given relaxation parameter λ value, the addition of EO stabilization parameter leads to faster convergence as it can be seen in Fig. 7.11. The number of iterations required for convergence significantly reduces to 51 from 301 iterations with the addition of EO stabilization at relaxation parameter $\lambda = 0.1$. This means similar to the EVSS formulation, the addition of the EO stabilization parameter helps at the convergence level and do not affect the solution atleast on the visual basis.

λ	$\gamma_{\boldsymbol{T}} = 0.0$	$\gamma_{\boldsymbol{T}} = 0.01$
0.01	10	9
0.1	301	51
0.2	–	60
0.5	–	192
0.6	–	471

Table 7.7: **Limits of λ for Oldroyd-B model:** Number of iterations required for convergence for increasing values of λ, and EO stabilization parameter $\gamma_{\boldsymbol{T}} = 0.0$ and $\gamma_{\boldsymbol{T}} = 0.01$.

Fig. 7.12 is prepared to show the components to stress tensor $\boldsymbol{T}$ at relaxation parameter $\lambda = 0.6$. It can be observed that the stress component T_{11}, T_{22} and T_{21}, takes the maximum

value at a transition from 4 units to 1 unit along the symmetry line. Component T_{12} takes the minimum value along the upper wall of the downstream channel i.e., $y = 1$.

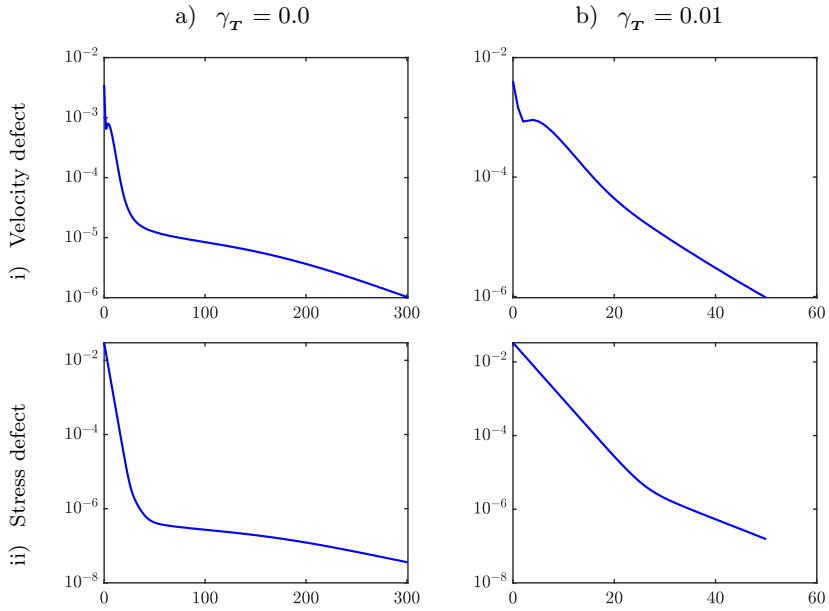


Fig. 7.11: **Convergence plots Oldroyd-B model:** Number of iterations required for convergence of the Oldroyd-B model with relaxation parameter value $\lambda = 0.1$, and EO stabilization parameter $\gamma_{\boldsymbol{T}} = 0.0$ and $\gamma_{\boldsymbol{T}} = 0.01$.

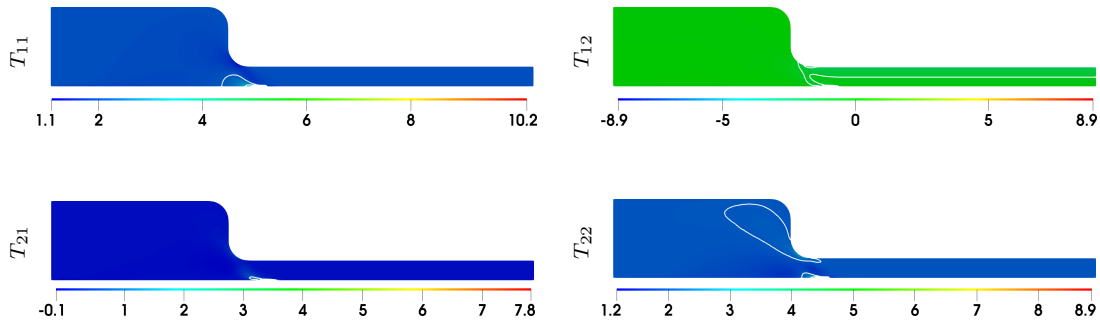


Fig. 7.12: **Oldroyd-B contraction flow:** Components of tensor $\boldsymbol{T}$ using Q_3 for the Oldroyd-B model at relaxation parameter $\lambda = 0.6$, and EO stabilization parameter $\gamma_{\boldsymbol{T}} = 0.01$ at mesh refinement level 3.

In the upcoming subsection, the results for the more complicated viscoelastic Giesekus model are simulated.

7.2.2 Giesekus model

The present subsection investigates the monolithic Tensor Stokes formulation for the Giesekus model for different values of the relaxation parameter λ. The results are obtained with and without the presence of the EO stabilization parameter in the constitutive equation. The Tensor Stokes formulation for the Giesekus model is solved subject to the following boundary conditions:

- On the left and right edge the semi-analytic 1-D velocity profiles are prescribed. The velocity profile for the zero solvent viscosity η_s=0 is an analytic function [21] that depends upon the material parameters and the pressure drop.

$$u_x(y) = \frac{B}{\lambda}\Big[\left(1-2\beta^2\right) log\frac{\beta+\sqrt{1-\phi^2\frac{y^2}{B^2}}}{\beta+\sqrt{1-\phi^2}} + \beta\left(\sqrt{1-\phi^2\frac{y^2}{B^2}}-\sqrt{1-\phi^2}\right) + \beta\left(1-\beta^2\right)\left(\frac{1}{\beta+\sqrt{1-\phi^2\frac{y^2}{B^2}}}-\frac{1}{\left(\beta+\sqrt{1-\phi^2}\right)}\right)\Big]/\phi \tag{7.13}$$

  where $\beta = 2\alpha - 1$ and $\phi = 2\alpha\lambda P_x$. The $y-$component of the velocity is zero $u_y = 0$.

- On the upper and lower wall set no slip boundary condition and symmetry boundary conditions, respectively.

- On the left edge the semi-analytic stress $\boldsymbol{T}$ profile is prescribed obtained by the stress decomposition (2.20).

$$\boldsymbol{T} = \begin{pmatrix} \frac{2\lambda P_x y}{B}\frac{1}{\partial u/\partial y} & \frac{(1-\alpha)\left(1\mp\sqrt{1-4\alpha^2\frac{\lambda^2P_x^2y^2}{B^2}}\right)+2\alpha^2\left(\frac{\lambda P_x y}{B}\right)^2}{\alpha\left(2\alpha-1\pm\sqrt{1-4\alpha^2\frac{\lambda^2P_x^2y^2}{B^2}}\right)}\frac{2}{\partial u/\partial y} \\ \frac{-1\pm\sqrt{1-4\alpha^2\frac{\lambda^2P_x^2y^2}{B^2}}}{2\alpha}\frac{2}{\partial u/\partial y} & \frac{2\lambda P_x y}{B}\frac{1}{\partial u/\partial y} \end{pmatrix} \tag{7.14}$$

  where,

$$\frac{\partial u}{\partial y} = \frac{2\alpha\lambda P_x y}{B}\frac{1\pm(2\alpha-1)\sqrt{1-\frac{4\alpha^2\lambda^2P_x^2y^2}{B^2}}}{\left(2\alpha-1\pm\sqrt{1-\frac{4\alpha^2\lambda^2P_x^2y^2}{B^2}}\right)^2}$$

where $\beta = 2\alpha - 1$ and $\phi = 2\alpha\lambda P_x$. Table 7.8 illustrates the importance of the EO stabilization parameter $\gamma_{\boldsymbol{T}}$ in the constitutive equation. The addition of EO stabilization to the constitutive equation helps get results at higher relaxation parameter λ as compared to no EO stabilization.

λ	$\gamma_{\boldsymbol{T}} = 0.0$	$\gamma_{\boldsymbol{T}} = 0.01$
0.01	18	28
0.1	297	33
0.2	–	44
0.8	–	553

Table 7.8: **Limits of λ for Giesekus model**: Number of iterations required for convergence at mesh refinement level 3 for Giesekus model at $\alpha = 0.1$.

The incorporation of EO stabilization speeds up convergence for a given λ value, as seen by the convergence charts in Fig. 7.13. Fig. 7.14 shows the components of the stress tensor

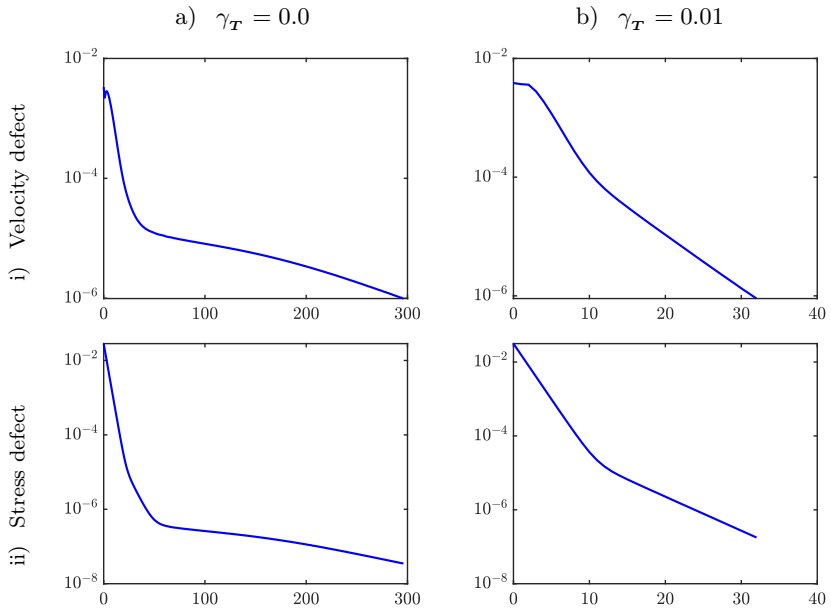


Fig. 7.13: **Convergence plots Giesekus model:** Number of iterations required for convergence at $\alpha = 0.1$ for increasing values of λ, and EO stabilization parameter $\gamma_{\boldsymbol{E}} = 0.0$ and $\gamma_{\boldsymbol{E}} = 0.01$.

$\boldsymbol{T}$ at relaxation parameter $\lambda = 0.8$.

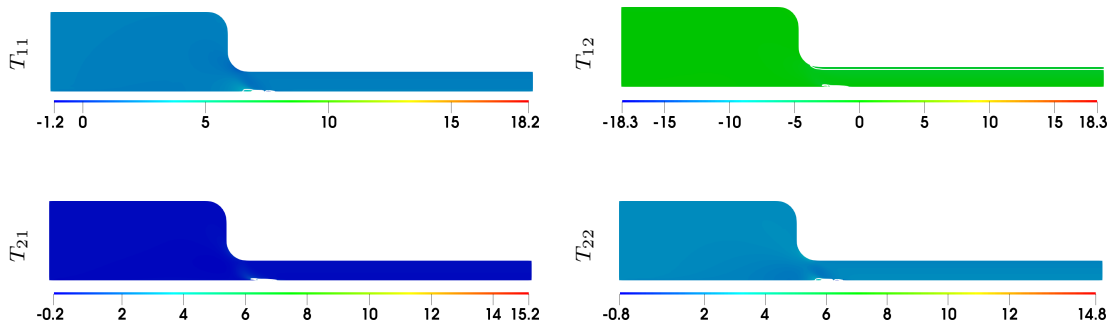


Fig. 7.14: **Giesekus model contraction flow:** Components of tensor $\boldsymbol{T}$ using Q_3 for Giesekus model at relaxation parameter value $\lambda = 0.8$, and EO stabilization parameter $\gamma_{\boldsymbol{T}} = 0.01$ at mesh refinement level 3.

7.2.3 PTT exponential model

Let us move forward with our investigation and check the affect of EO stabilization parameter on the PTT exponential model. The results are prepared for increasing value of the relaxation parameter λ. The Tensor-Stokes formulation for the PTT-exponential model is solved subject to the following boundary conditions:

- On the left and right edge the semi-analytic 1-D velocity profiles are prescribed. The velocity profile for the zero solvent viscosity $\eta_s = 0$ is an analytic function [21] that depends upon the material parameters and the pressure drop.

$$u_x(y) = \frac{\eta_p}{4\kappa\lambda^2\frac{\partial p}{\partial x}}\left[exp\left(2\kappa\left(\frac{\lambda}{\eta_p}\frac{\partial p}{\partial x}\left(y - \frac{b+a}{2}\right)\right)^2\right) - exp\left(2\kappa\left(\frac{\lambda}{\eta_p}\frac{\partial p}{\partial x}\left(\frac{b-a}{2}\right)\right)^2\right)\right] \tag{7.15}$$

  and $u_y = 0$.

- On the upper and lower wall set no slip and symmetry boundary condition, respectively.

- On the left edge, the semi-analytic stress tensor $\boldsymbol{T}$ computed by employing extra-stress tensor $\boldsymbol{S}$ obtained from [21] is given as,

$$\boldsymbol{T} = \begin{pmatrix} \frac{2\eta_p}{exp\left(2\kappa\left(\frac{\lambda}{\eta_p}\frac{\partial p}{\partial x}y\right)^2\right)} & \frac{4\eta_p\lambda y\frac{\partial P}{\partial x}}{exp\left(2\kappa\left(\frac{\lambda}{\eta_p}\frac{\partial p}{\partial x}y\right)^2\right)} \\ 0 & \frac{2\eta_p}{exp\left(2\kappa\left(\frac{\lambda}{\eta_p}\frac{\partial p}{\partial x}y\right)^2\right)} \end{pmatrix} \tag{7.16}$$

Again, as expected the addition of the EO stabilization parameter $\gamma_{\boldsymbol{T}}$ to the stress component to the constitutive equation largely helps in obtaining convergence at high relaxation parameter λ. This behavior can be seen in Table 7.9. Also, the accelerated convergence for smaller values of the relaxation parameter λ with the addition of EO stabilization parameter can be seen in Fig. 7.15. Fig. 7.16 is prepared to show the graphical visualization of the components of the stress tensor at $\lambda = 0.8$.

λ	$\gamma_{\boldsymbol{T}} = 0.0$	$\gamma_{\boldsymbol{T}} = 0.01$
0.01	29	29
0.10	303	33
0.20	–	40
0.50	–	120
0.80	–	546

Table 7.9: **Limits of λ for PTT exponential model**: Number of iterations required for convergence at $\kappa = 0.1$ for increasing values of the relaxation parameter λ, and EO stabilization parameter $\gamma_{\boldsymbol{T}} = 0.0$ and $\gamma_{\boldsymbol{T}} = 0.01$.

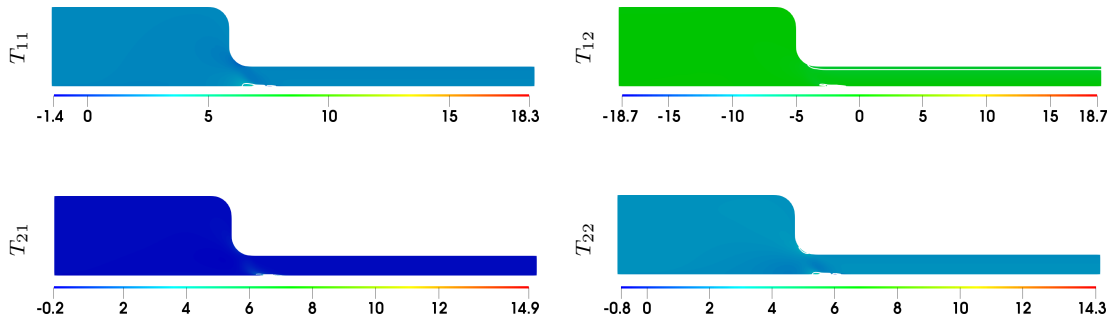


Fig. 7.16: **PTT exponential contraction flow:** Components of tensor $\boldsymbol{T}$ using $Q3$ for PTT exponential model at relaxation parameter $\lambda = 0.8$, and EO stabilization parameter $\gamma_{\boldsymbol{T}} = 0.01$ at mesh refinement level 3.

Having tested the effects of EO stabilization in detail for the EVSS and Tensor Stokes formulation on the constitutive equation. Our main conclusion is that the addition of the EO stabilization in the constitutive equation helps us to achieve the results at high relaxation parameter λ for both the formulations and three viscoelastic models. We also

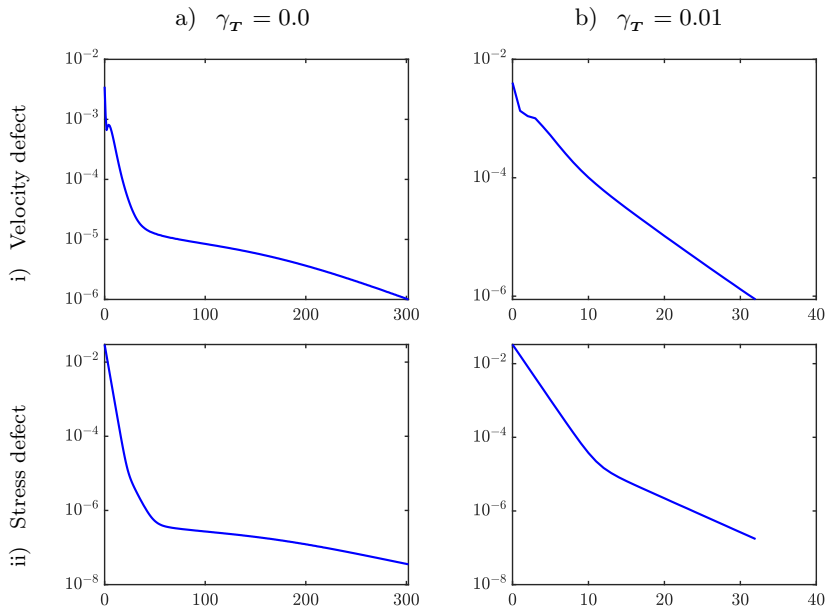


Fig. 7.15: **Convergence plots PTT exponential model:** Number of iterations required for convergence at relaxation parameter value $\lambda = 0.1$, and EO stabilization parameter $\gamma_{\boldsymbol{T}} = 0.0$ and $\gamma_{\boldsymbol{T}} = 0.01$.

saw that for lower value of the relaxation parameter λ, the application of EO stabilization aids in the faster convergence.

7.3 Comparison between EVSS and Tensor Stokes formulation in terms of extra stress tensor

In this section, a comparison between the EVSS and Tensor Stokes formulations regarding the extra stress tensor is provided. The components of the extra stress tensor $\boldsymbol{S}$ obtained from the post processing of the EVSS and Tensor Stokes formulations for the Oldroyd-B fluid are presented. These components of tensor $\boldsymbol{S}$ are acquired utilizing the stress splitting in equation (2.11) and (2.20). The graphical representation of the components of tensor $\boldsymbol{S}$ at relaxation parameter $\lambda = 0.5$ is depicted in Fig. 7.17. It is notable that the extra-stress tensor obtained from both formulations exhibits good agreement on the visual basis.

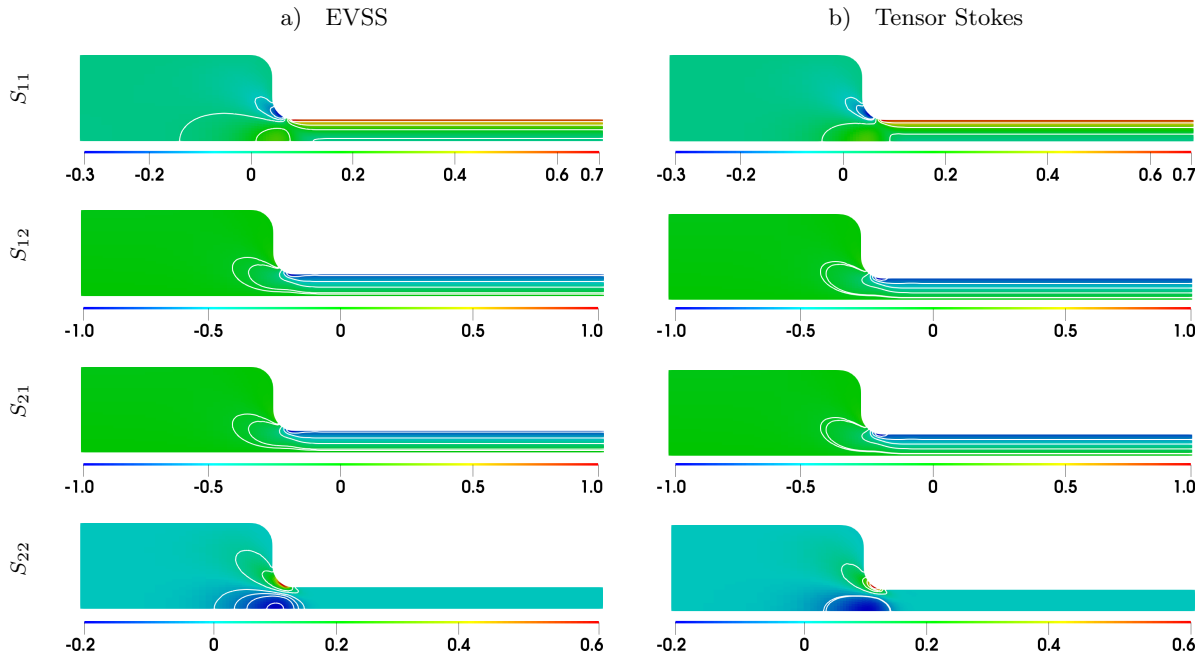


Fig. 7.17: **Comparative analysis of EVSS and Tensor Stokes formulation:** Comparison of the components of extra stress tensor $\boldsymbol{S}$ obtained by the post processing of the EVSS and Tensor Stokes results at relaxation parameter $\lambda = 0.5$ at mesh refinement level 3.

The next section deals in finding the limits of the relaxation parameter λ with the application of the EO stabilization parameter in the momentum equation and presents a comparison of the computational cost between the decoupled and monolithic solution approaches for the EVSS formulation.

7.4 Computational cost for the decoupled and monolithic EVSS formulation

In this section, we will continue only with the EVSS formulation with the addition of the EO stabilization to the momentum equation for the decoupled and monolithic formulation for the viscoelastic models of interest to check the limits of the relaxation parameter λ. A comparison of the computational cost between the decoupled and monolithic formulation is also presented in this section.

7.4.1 Oldroyd-B fluid

The numerical investigation starts with the basic Oldroyd-B model, utilizing the boundary conditions outlined in section 7.1.1 for the current computations. In the current setup, the limits of the relaxation parameter λ are documented using both decoupled and monolithic solution approaches. As the relaxation parameter λ increases, intensifying the viscoelastic effects, the problem becomes more challenging to solve. The computations are performed on compute server with Intel Xeon E5-2640 v3 with 16 cores and 128 GiB DDR4 Memory. Among the two approaches, the decoupled formulation proves to be less costly in terms of time and memory, as demonstrated in Table 7.10.

λ	Decoupled			Monolithic		
	Iterations	Run Time (sec)	Memory (GB)	Iterations	Run Time (sec)	Memory (GB)
0.0	2	08	0.951	2	13	1.551
0.5	29	167	1.119	5	81	1.787
1.0	59	334	1.121	15	243	1.834
3.0	181	1048	1.131	86	2048	2.100
5.0	1091	6222	1.129	-	-	-

Table 7.10: **Computational cost using EVSS formulation:** Computational cost for the convergence of the Oldroyd-B model at increasing values of the relaxation parameter λ at mesh refinement level 3 and $\gamma_{\mathbf{u}} = 1$.

The monolithic approach is more robust because it can solve for all field variables simultaneously. This robustness is evident in the convergence plots shown in Fig. 7.18, where the EO stabilization parameter $\gamma_{\mathbf{u}}$ has little to no influence on the velocity and stress convergence plots obtained by the monolithic formulation while the decoupled approach is heavily influenced by the addition of the stabilization. However, no convergence could be achieved at relaxation parameter $\lambda = 5$ for the monolithic formulation. Keeping in mind, Oldroyd-B fluid is the simplest viscoelastic model let us proceed towards somewhat more complicated viscoelastic model i.e., Giesekus model in the next subsection.

7.4.2 Giesekus Model

A comparison of the computational cost between decoupled and monolithic formulation for the Giesekus model is prepared in the following section. Again we have utilized the

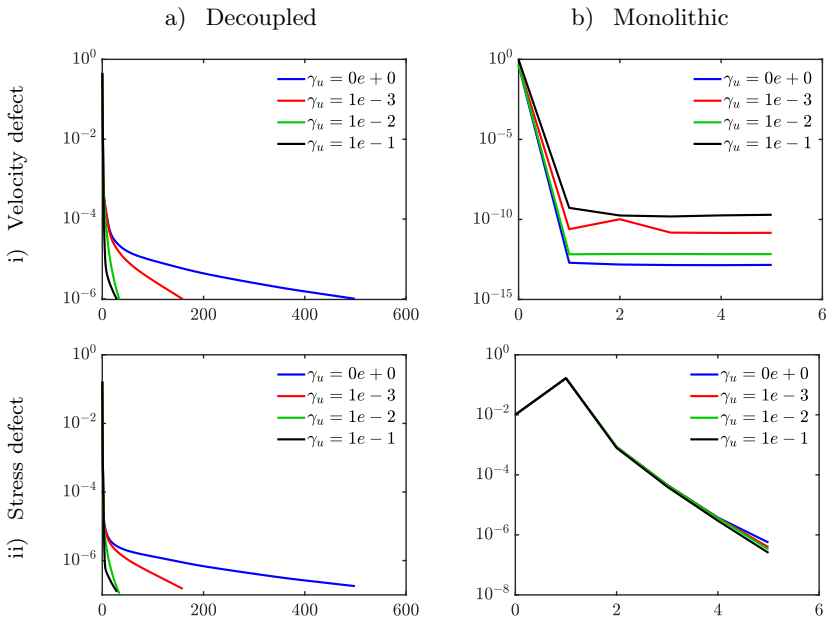


Fig. 7.18: **Effects of EO stabilization on Oldroyd-B model:** Effects of EO stabilization parameter $\gamma_{\mathbf{u}}$ on the number of iterations required for convergence of the Oldroyd-B model at relaxation parameter value $\lambda = 0.5$ at mesh refinement level 3.

boundary conditions for the Giesekus model outlined in section 7.1.2 for the current computations. Like the Oldroyd-B model, the monolithic formulation for the Giesekus model is more robust and unaffected by the stabilization parameter, as illustrated in Fig. 7.19. In the decoupled formulation, increasing the stabilization parameter $\gamma_{\mathbf{u}}$ aids in achieving velocity and stress convergence in fewer iterations, thereby facilitating overall convergence. In contrast, the monolithic formulation's convergence appears to be independent of $\gamma_{\mathbf{u}}$, at least for smaller values of the relaxation parameter λ.

Table 7.11 illustrates that the decoupled formulation outperforms the monolithic approach in terms of the computational cost. The increased robustness of the monolithic approach comes at the cost of a more expensive solution for the resulting coupled discrete nonlinear system. Although the decoupled/operator-splitting scheme simplifies the overall solution into a sequence of much simpler subproblems, the outer coupling of these subproblems remains a challenging task, particularly when dealing with a high relaxation parameter λ for models like Giesekus and PTT exponential.

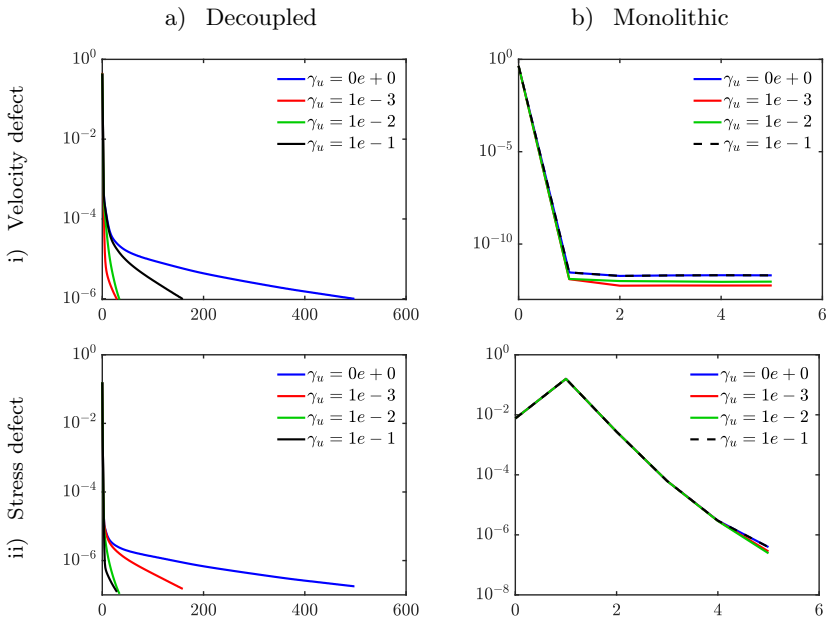


Fig. 7.19: **Effects of EO stabilization on Giesekus model:** Effects of EO stabilization parameter $\gamma_{\mathbf{u}}$ on the number of iterations required for convergence of the Giesekus model at relaxation parameter value $\lambda = 0.5$.

λ	Decoupled			Monolithic		
	Iterations	Run Time (sec)	Memory (GB)	Iterations	Run Time (sec)	Memory (GB)
0.1	6	40	1.093	4	45	1.699
0.5	29	170	1.117	5	82	1.783
1.0	59	340	1.121	7	122	1.835
3.0	168	974	1.127	53	1279	2.109
5.0	345	1986	1.127	186	4804	2.160

Table 7.11: **Computational cost using EVSS formulation:** Computational cost for the convergence of the Giesekus model for increasing values of the relaxation parameter λ at mesh refinement level 3 and $\gamma_{\mathbf{u}} = 1$.

A similar numerical solution technique is applied for the PTT exponential model. PTT exponential model yields numerical solutions akin to those of the Giesekus model. Nonetheless, for the sake of completeness, the subsequent section provides a brief analysis of the PTT exponential model.

7.4.3 PTT exponential model

The flow profiles for the PTT exponential model are evaluated with respect to the effect of the included material parameter on the nonlinearity of the solution. Again, we have utilized the boundary conditions for the PTT exponential model outlined in section 7.1.3 for the current computations. The computational cost for the PTT exponential model is given in Table 7.12.

λ	Decoupled			Monolithic		
	Iterations	Run Time (sec)	Memory (GB)	Iterations	Run Time (sec)	Memory (GB)
0.1	6	40	1.096	3	64	1.749
0.5	29	182	1.114	7	103	1.714
1.0	59	369	1.114	15	321	2.034
3.0	150	927	1.128	66	1539	2.095
10.0	962	5849	1.132	-	-	-

Table 7.12: **Computational cost using EVSS formulation:** Computational cost for the convergence of the PTT exponential model at mesh refinement level 3 and $\gamma_{\mathbf{u}} = 1$.

Identical convergence behavior to the Oldroyd-B and Giesekus model is observed for the PTT exponential model and is therefore not included. The decoupled PTT exponential approach provides a more realistic representation of viscoelastic effects, notably achieving accurate results even with very high relaxation parameter, such as $\lambda = 10$. It would be interesting to visualize the flow behavior specially near the contraction in Fig. 7.20 at such high relaxation parameter λ. The stress component E_{11} and $E_{12} = E_{21}$ are maximum near the upper wall of the downstream channel i.e., $y = 1$. The component E_{22} gets maximum at a transition from 4 to 1 unit along the symmetry line.

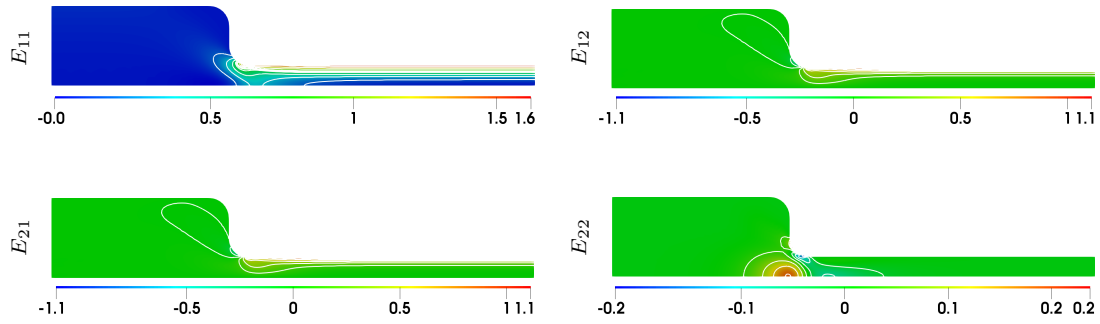


Fig. 7.20: **PTT exponential contraction flow:** Components of tensor $\boldsymbol{E}$ using Q_3 for PTT exponential model at relaxation parameter $\lambda = 10$ at mesh refinement level 3.

The success of obtaining results at high relaxation parameter λ has motivated us to pursue further work with PTT exponential model using a decoupled routine. Our objective is to develop an algorithm with such solver settings that facilitate efficient and realistic predictions of viscoelastic behavior, without encountering any limitations in the relaxation parameter λ.

7.5 Non-stationary decoupled PTT exponential EVSS formulation

Until now, the primary focus has been on obtaining numerical solutions for the steady-state viscoelastic flow problems. In this section, we aim to work on the decoupled EVSS formulation using the PTT exponential viscoelastic model, with the goal of operating in a non-stationary environment to achieve steady-state solutions at higher values of the relaxation parameter λ. This shift is necessary because, when dealing with complex rheological flows and aiming to obtain results for highly viscoelastic flows (i.e., flows with a high relaxation parameter λ), a time-dependent approach may be the only accurate method to compute the steady-state solution. At each time step a non-linear problem in $\mathbf{u}, p$ and $\boldsymbol{E}$ needs to be solved in a decoupled way. The time dependent terms are discretized using Implicit Euler method. The computations for the subproblems i.e., the Stokes and constitutive equation are performed using UMFPACK and FGMRES iterative solver, respectively. To ensure convergence at each time step, the relative changes in velocity and stress are computed, with a tolerance threshold of 10^{-6}. The relative changes are calculated as follows:

Relative change in velocity $= \frac{||\mathbf{u}^n - \mathbf{u}^{n-1}||}{\delta_t} <$TOL

Relative change in stress $= \frac{||\boldsymbol{E}^n - \boldsymbol{E}^{n-1}||}{\delta_t} <$TOL

In upcoming subsection, we briefly summarize the time discretization techniques.

7.5.1 Adaptive time stepping technique

Having developed the non-stationary environment for the viscoelastic computations it is realized that the numerical simulations were quite costly in terms of number of steps and eventually the time required to achieve the steady state solution. An adaptive time stepping technique is introduced such that the initial time step size, δ_t, is small. As the algorithm begins to converge, δ_t is gradually increased, resulting in accelerated convergence and the development of more efficient algorithm. The proposed adaptive time stepping technique works on the following principle,

$$\begin{gathered} \text{Relative change in velocity} < \text{certain tolerance} \\ \text{And} \\ \text{Relative change in stress} < \text{certain tolerance} \\ => \delta_t = \alpha \delta_t \end{gathered} \tag{7.17}$$

where α is constant. Fig. 7.21 - 7.25 represents the convergence behavior of velocity and stress with constant time stepping size δ_t and proposed adaptive time stepping technique (7.17) for increasing values of the relaxation parameter λ. The influence of time step δ_t is also visualized on the number of the FGMRES iterations apart from its effects on the velocity and stress convergence. In Fig. 7.21, at relaxation parameter $\lambda = 1$, one could see that an increase in the time step δ_t does not affect the number of FGMRES iterations for an adaptive time stepping technique. However, the velocity and stress convergence is highly accelerated by the use of adaptive time stepping over the constant time stepping. Fig. 7.22 is prepared employing the constant time stepping and adaptive time stepping technique at relaxation parameter $\lambda = 5$. Again one could see the supremacy of the adaptive time stepping technique over the constant time stepping. Also, it is worth noticing that at high relaxation parameter $\lambda = 5$, the increase in δ_t not only affect the convergence behavior but also the number of the FGMRES iterations. As the time step size δ_t increase, the number of iterations required by the FGMRES solver also tends to increase, particularly at high δ_t values. However, after a few iterations, the number of FGMRES iterations starts to decrease. This phenomenon suggests that as the algorithm converges at high δ_t values, the FGMRES solver becomes more efficient, requiring fewer iterations even as the time step δ_t continues to increase.

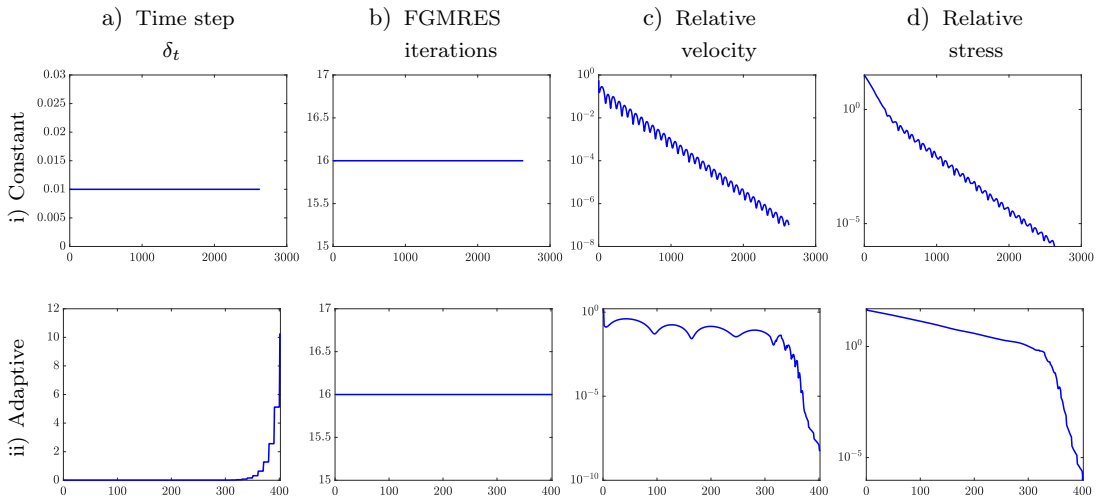


Fig. 7.21: **Non stationary PTT exponential fluid flow:** Effects of constant time step step size $\delta_t = 0.01$ and adaptive time stepping on number of FGMRES iterations and relative velocity and stress at $\lambda = 1$.

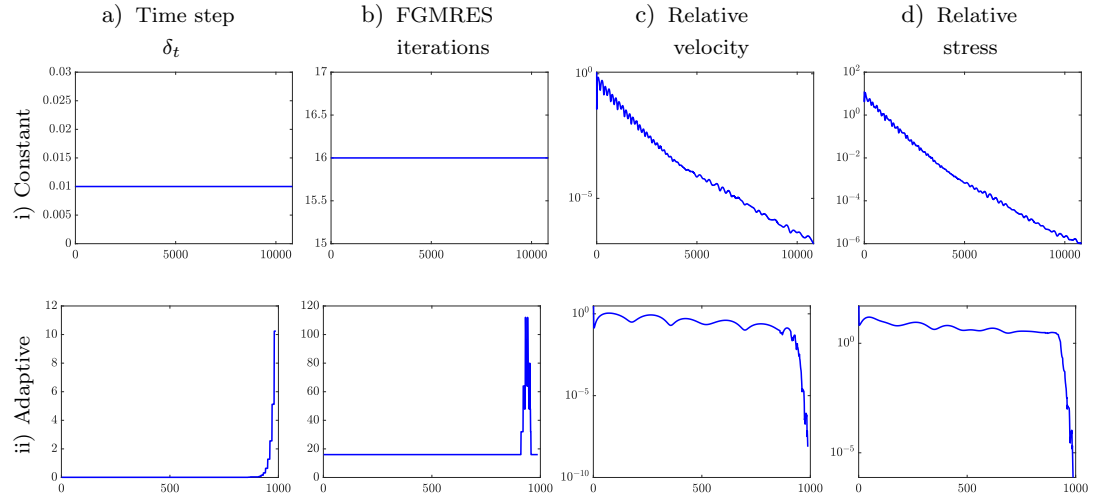


Fig. 7.22: **Non stationary PTT exponential fluid flow:** Effects of constant time step step size $\delta_t = 0.01$ and adaptive time stepping on number of FGMRES iterations and relative velocity and stress at $\lambda = 5$.

The adaptive time stepping technique helps to achieve the results for highly viscoelastic flows i.e., in the current setting, we encountered no limits of the relaxation parameter λ. The numerical simulations were intentionally stopped after $\lambda = 50$. The effect of adaptive

time stepping technique for the relaxation parameter $\lambda = 10,\ 20$ and 50 on the FGMRES solver and velocity and stress convergence are shown in Fig. 7.23, 7.24 and 7.25.

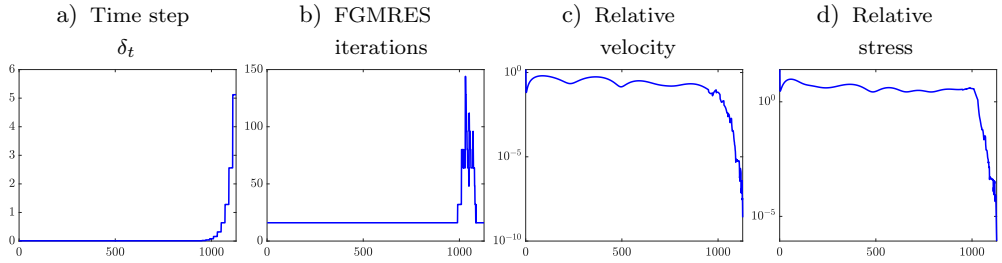


Fig. 7.23: **Non stationary PTT exponential fluid flow:** Effects of adaptive time stepping on number of FGMRES iterations and relative velocity and stress at relaxation parameter $\lambda = 10$.

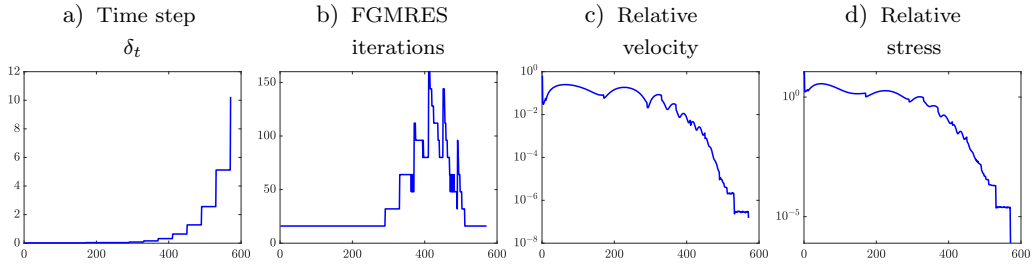


Fig. 7.24: **Non stationary PTT exponential fluid flow:** Effects of adaptive time stepping on number of FGMRES iterations and relative velocity and stress at relaxation parameter $\lambda = 20$.

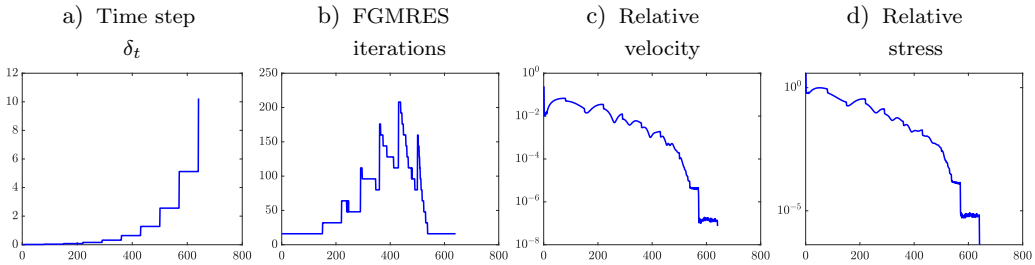


Fig. 7.25: **Non stationary PTT exponential fluid flow:** Effects of adaptive time stepping on number of FGMRES iterations and relative velocity and stress at relaxation parameter $\lambda = 50$.

Fig. 7.26 is prepared to show flow solutions for the PTT exponential model at high relaxation parameter $\lambda = 50$. The horizontal velocity depicted in Fig. 7.26 deviates from the typical parabolic profile due to the shear thinning effect. Initially, it exhibits a maximum value aligned with the symmetry line at the start of the downstream channel. However, as the flow progresses, it transitions into a plateau-like behavior, deviating from the parabolic shape. The stress component contours E_{11} and E_{12} exhibit maximum values near the curved corner contraction, where the geometry undergoes a sudden reduction from 4 units to 1 unit. Maximum values of the stress are also noticeable along the channel wall. This behavior is in accordance with the stress function provided earlier. The stress component E_{22} exhibits a contour line pattern with its maximum value near the curved corner contraction and along the symmetry line, where the fluid undergoes a sudden change in geometry.

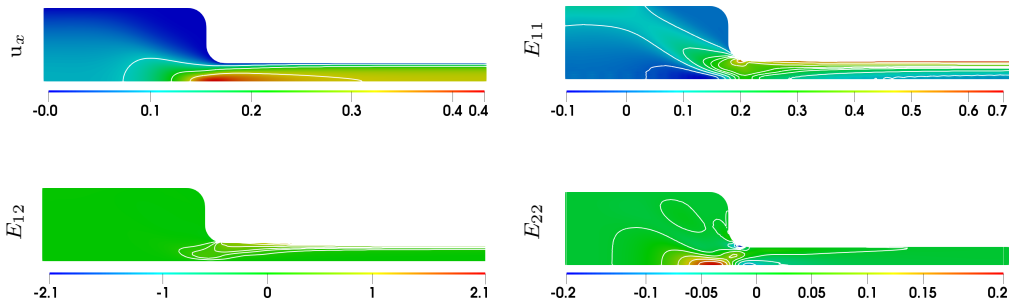


Fig. 7.26: **PTT exponential contraction flow:** Components of tensor $\boldsymbol{E}$ using Q_3 for PTT exponential model at relaxation parameter $\lambda = 50$ at mesh refinement level 3.

Our aim to obtain the results for highly viscoelastic fluid i.e., at high relaxation parameter λ without restricting λ is accomplished, using the adaptive time stepping technique. In the next section, let us briefly modify the adaptive time stepping technique to make it more efficient.

7.5.2 Modified adaptive time stepping technique

The adaptive time stepping technique described in above section is designed to obtain the steady state solution of the highly viscoelastic three field EVSS formulation. The primary objective of obtaining results at extremely high relaxation parameter λ has been successfully accomplished without encountering any limitation in the relaxation parameter value λ. However, it was observed that for highly viscoelastic fluids, specifically with $\lambda = 50$, the number of FGMRES iterations increases significantly as the time step δ_t is increased, surpassing 200 iterations. Although the number of FGMRES iterations begins to decrease

after a certain point, we aim to minimize such high number of FGMRES iterations. Therefore, a modification to the adaptive time stepping technique is performed.
The algorithm begins by initializing the solution variables $\mathbf{u}$ and $\boldsymbol{E}$ for the next approximate solution based on the current time level. The modified adaptive time stepping technique dynamically adjusts the time step δ_t based on the number of the iterations required by FGMRES solver. The algorithm for the modified adaptive time stepping technique is described below as follows:

Algorithm 1 Modified adaptive time stepping technique

Let $(\mathbf{u}, p, \boldsymbol{E})^n$ be the solution at n^{th} time level
if $(\mathbf{u}, p, \boldsymbol{E})^{(n+1)} \approx (\mathbf{u}, p, \boldsymbol{E})^{(n)}$ & FGMRES iterations $< \mathbb{N}$ **then**
    Set $\delta_t = \alpha \delta_t$ ▷ $\alpha > 1$
else if FGMRES iterations $> \mathbb{N}$ **then**
    Set $\delta_t = \alpha \delta_t$ ▷ $0 < \alpha < 1$
else
    Set $\delta_t = \delta_t$
end if

The numerical results are computed at a higher relaxation parameter $\lambda = 50$, utilizing the initial solution obtained from the previous iteration at a lower relaxation parameter of $\lambda = 49$. Table 7.13 shows the effect of the maximum number of the FGMRES iterations on the number of steps required to obtain the convergence. Fig. 7.27 shows the influence of the maximum number of FGMRES iterations on the time step δ_t and the convergence plots for the velocity and stress using the modified adaptive time stepping technique. It can be noted that when the maximum number of FGMRES iterations is capped at 90, an increase in the time step δ_t leads to a corresponding increase in the number of FGMRES iterations. However, once the maximum threshold of 90 iterations is reached, the time step δ_t begins to decrease, consequently reducing the number of FGMRES iterations. This iterative process continues until convergence is achieved, adhering to the imposed restriction on the maximum number of FGMRES iterations. Well, we can control the number of FGMRES iterations to some extent by the modified adaptive time stepping technique. However, if the number of FGMRES iterations is capped to 70, then no convergence could be achieved. Additionally, it is necessary to keep a balance, as imposing a limit on the number of FGMRES iterations results in an increase in the overall number of steps required for convergence.

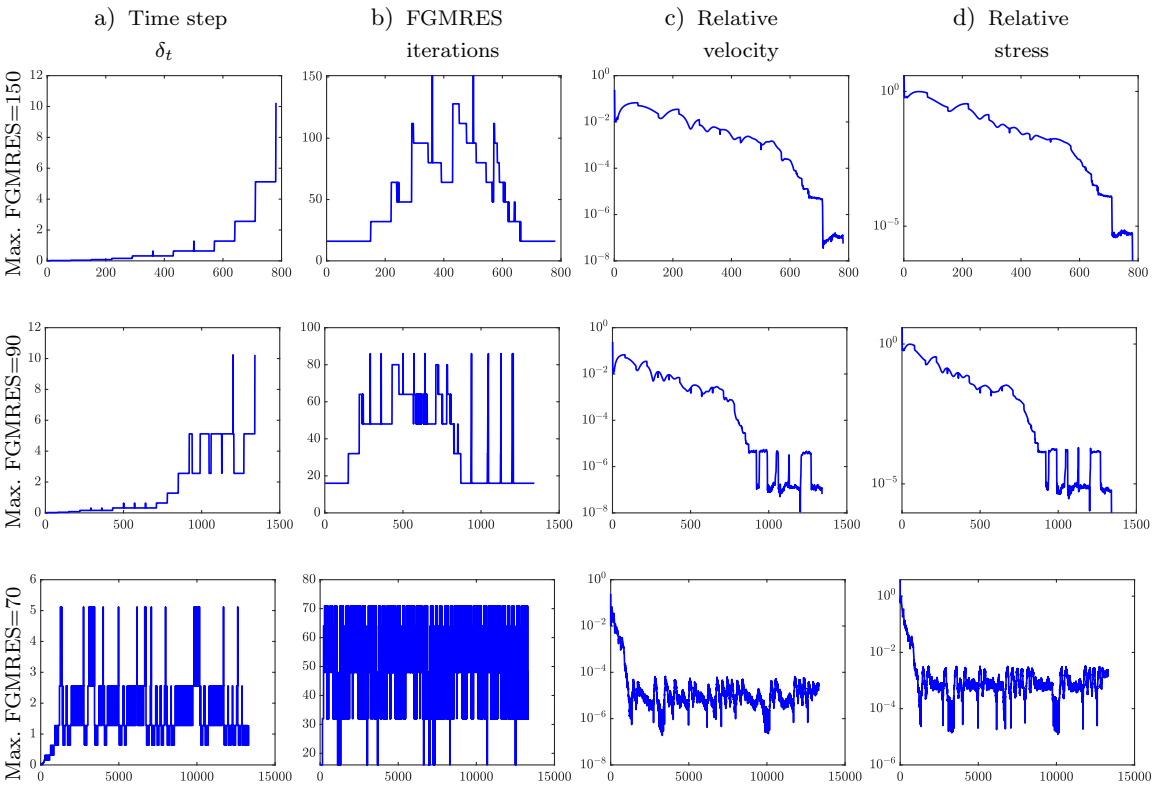


Fig. 7.27: **Modified adaptive time stepping technique:** Effects of maximum number of FGMRES iterations on step size δ_t and convergence at relaxation parameter $\lambda = 50$

Maximum number of FGMRES iterations	Number of steps
210	641
150	781
130	782
100	1062
90	1341
70	Relative changes stagnates

Table 7.13: **Modified adaptive time stepping technique:** Effect of maximum number of FGMRES iterations on the number of steps required to meet the tolerance at $\lambda = 50$.

In the next section, let us briefly study the Multigrid performance on the decoupled three field EVSS formulation using PTT exponential model.

7.6 Multigrid solver performance

In the previous section, the results for highly viscoelastic fluids i.e., high relaxation parameter $\lambda = 50$ were obtained for the decoupled non-stationary formulation. The Stokes subproblem was solved using the direct solver UMFPACK while FGMRES was used to obtain results for the constitutive equation for stress. In this section, let us replace the direct solver UMFPACK for the Stokes flow by the multigrid solver while keeping FGMRES solver for the constitutive equation. The Stokes problem in $\mathbf{u}$ and p is solved using multigrid with Vanka as a preconditioner to the FGMRES smoother. Fig. 7.28 is plotted to show how multigrid iterations and convergence behaves for increasing values of the relaxation parameter λ and how it affects the velocity and stress convergence. It can be seen that at multigrid works nicely until relaxation parameter $\lambda = 5$ but at relaxation parameter $\lambda = 6$ multigrid fails to provide convergence.

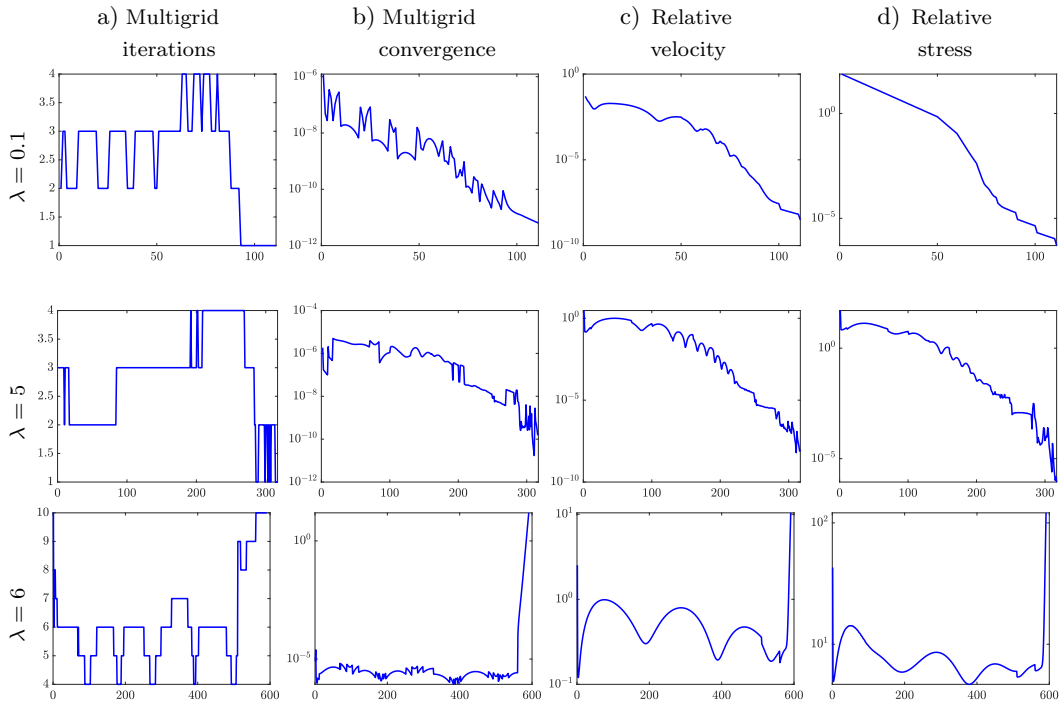


Fig. 7.28: **Viscoelastic multigrid solver:** Effect of number of multigrid iterations and its convergence behavior on the relative velocity and stress for increasing values of λ.

Let us analyze the possible reasons for the limitation of the Multigrid solver to provide convergence at higher values of the relaxation parameter λ.

Effect of EO stabilzation on multigrid solver performance:

Since, we are working with the decoupled formulation i.e., the Stokes part is solved separately from the constitutive equation. In order to find the reason why multigrid is unable to provide promising results for highly viscoelastic fluids i.e., for high relaxation parameter λ we simply ignored the constitutive equation for stress for some time and only focus on the Stokes problem with the constant right hand side for stress. The considered problem reads as,

$$\begin{aligned} \rho\frac{\partial \mathbf{u}}{\partial t} - 2\eta_0 \nabla \cdot \mathbf{D}(\mathbf{u}) + \nabla p &= \nabla \cdot \boldsymbol{E} \\ \nabla \cdot \mathbf{u} &= 0 \end{aligned} \tag{7.18}$$

where $\eta_0 = \eta_s + \eta_p = 1$ and $\rho = 1$ are the constants. The problem (7.18) is solved using a constant right hand side provided the semi-analytic $\boldsymbol{E}$ function given below:

$$\boldsymbol{E} = \begin{pmatrix} \frac{2\lambda}{\eta_p}\left(\frac{\partial p}{\partial x}y\right)^2 & \frac{\partial p}{\partial x}y\left(1 - exp\left(2\kappa\left(\frac{\lambda}{\eta_p}\frac{\partial p}{\partial x}y\right)^2\right)\right) \\ \frac{\partial p}{\partial x}y\left(1 - exp\left(2\kappa\left(\frac{\lambda}{\eta_p}\frac{\partial p}{\partial x}y\right)^2\right)\right) & 0 \end{pmatrix}$$

The numerical results for the Stokes flow with constant right hand side can be computed for any relaxation parameter λ without having to solve for the stress equation separately. Since, we are interested in high values of the relaxation parameter $\lambda = 50$ so we decided to check the effect of time step size δ_t and EO stabilization parameter $\gamma_{\mathbf{u}}$ on the convergence behavior of the Multigrid solver. The results are obtained on two mesh refinement levels, level 2 and level 3 with 992 and 3968 quadrilaterals, respectively.
Table 7.14 is presented to show the limits of the edge oriented stabilization parameter $\gamma_{\mathbf{u}}$ with increase in the time step size δ_t at mesh refinement level 2 and 3.

δ_t	EO stabilization parameter $\gamma_{\mathbf{u}}$	
	Level 2	Level 3
0.01	0.4	0.2
0.10	0.1	0.05
0.50	0.05	0.05
1.00	0.05	0.05

Table 7.14: **Multigrid solver performance:** Range of the EO stabilization $\gamma_{\mathbf{u}}$ with increase in δ_t at mesh refinement level 2 and 3 for relaxation parameter $\lambda = 50$.

The convergence plots for levels 2 and 3 with time step sizes $\delta_t = 0.01$ and 1 are shown in Fig. 7.29 and 7.30, respectively.

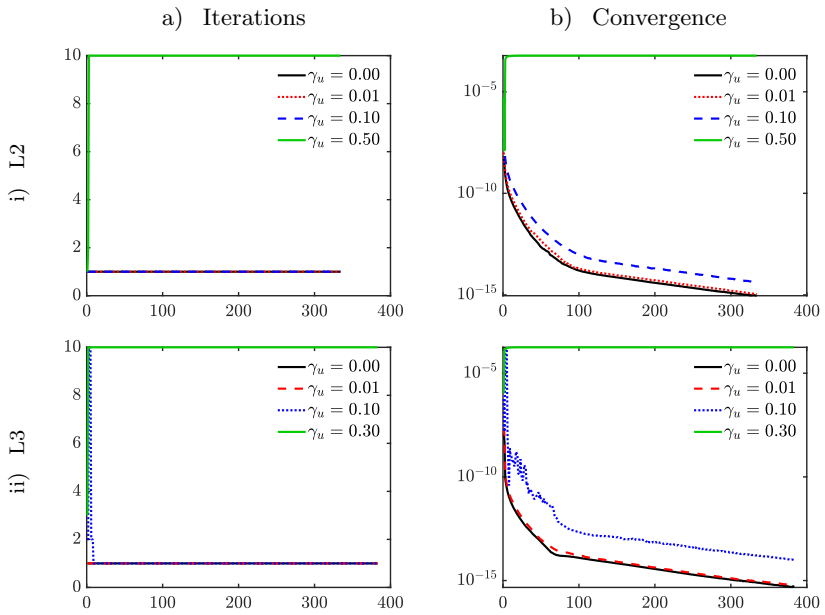


Fig. 7.29: **Multigrid solver performance:** Effect of EO stabilization $\gamma_{\mathbf{u}}$ on multigrid iterations and convergence at $\delta_t = 0.01$ and $\lambda = 50$ at mesh refinement level 2 and 3.

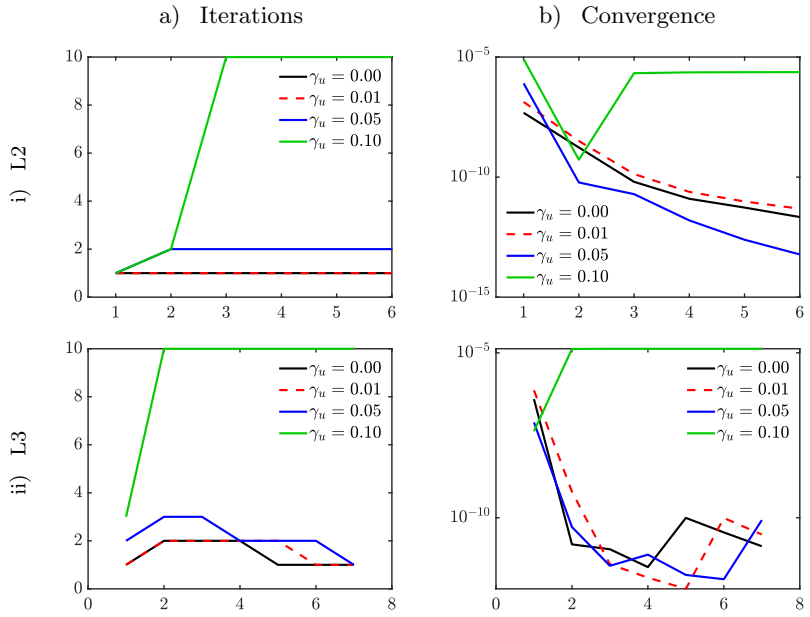


Fig. 7.30: **Multigrid solver performance:** Effect of EO stabilization $\gamma_{\mathbf{u}}$ on multigrid iterations and convergence at $\delta_t = 1$ and $\lambda = 50$ at mesh refinement level 2 and 3.

It can be observed in both the figures that at both refinement levels, 2 and 3, the multigrid solver fails to converge beyond a certain stabilization parameter, $\gamma_{\mathbf{u}}$. In conclusion, the multigrid solver can only achieve convergence for $\gamma_{\mathbf{u}} \leq 0.05$. However, to obtain convergence at higher relaxation parameters λ, EO stabilization with $\gamma_{\mathbf{u}} > 0.05$ is necessary. In the upcoming section, let us briefly visualize the shear thinning effect produced by viscoelastic Giesekus and PTT exponential model.

7.7 Shear thinning effect

Viscoelastic materials exhibit shear thinning behavior, characterized by a reduction in shear viscosity as shear rate is increased [46]. The flow profiles computed for both the Giesekus and PTT exponential models, within fully developed channel flow configurations, demonstrate reasonable behavior across different relaxation time λ. As relaxation times increase, Fig. 7.31 illustrates a typical shear thinning behavior observed in the velocity profiles. This behavior, a key material property predicted by both the Giesekus and PTT exponential models, is notable. During shear thinning, the velocity profile deviates from the parabolic profile, resulting in a large velocity gradient near the walls and a plateau-like behavior downstream of the channel is observed. As we increase the relaxation parameter, the plateau effect becomes more noticeable. Moreover, there is a high level of agreement with the semi-analytical reference solution provided in [21], as observed visually.

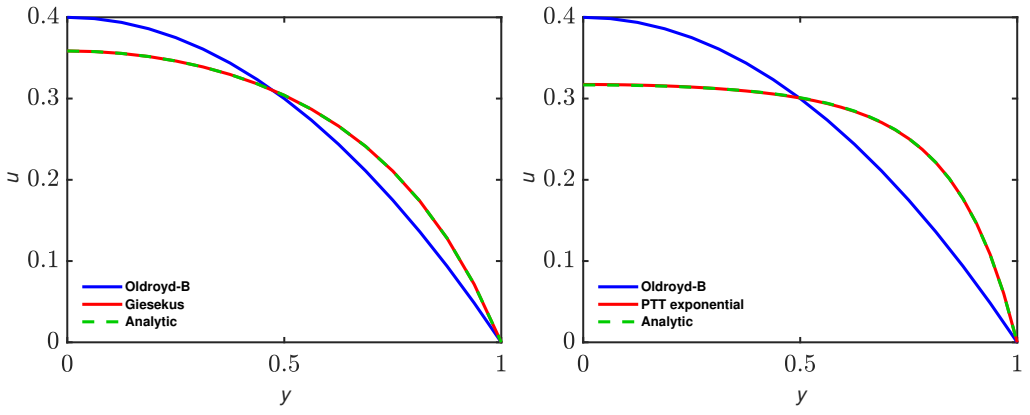


Fig. 7.31: **Fully developed viscoelastic flow:** Shear thinning effect produced by Giesekus and PTT exponential model at $\lambda = 5$ and 50, respectively at the downstream channel $x = 20$.

8

Conclusions

This research primarily aimed to develop and apply simulation techniques for viscoelastic fluid flow, with a specific emphasis on pure polymer melts. These viscoelastic fluids are characterized by the absence of any solvent contribution to the viscosity. As a result, the numerical difficulties and challenges that may arise are outlined by describing existing finite element techniques for simulating such types of fluids. The goal of this work is to gain a better understanding of the problem of numerically solving viscoelastic flow simulations.

The physical set of equations, which couple the generalized Stokes and constitutive equation for the stress, are introduced in the second chapter of the work that constitutes the differential viscoelastic model. When considering pure polymer melts, the solvent contribution to viscosity vanishes, as well as the velocity coupling into the momentum equation, making it difficult to handle the set of equations in a monolithic way specially in terms of application of the multigrid solver and making it impossible to solve the problem in a decoupled way without any added stabilization. The change of variables is performed into the original set of equations by employing the stress decomposition in two ways, resulting in the Elastic Viscous Stress Splitting (EVSS) and Tensor Stokes formulation. In the case of zero solvent viscosity, both formulations help in the recovery of the diffusive operator back into the momentum equation. On the one hand, having recovered the diffusive operator into the momentum equation definitely helps at the solver level, however the second-order velocity derivatives appear in the convective term of the constitutive equation in both the EVSS and Tensor Stokes formulations. To overcome this problem, the deformation tensor is typically treated as an additional variable. This increases the problem to four-field formulation thus leading to higher computational cost. In this thesis, a remedy to treat the second order velocity derivatives is presented. We explicitly transform the convective term while assuming that the velocity field is divergence-free eventually retaining the problem size to three field.

The potential advantages of stress splitting in the EVSS and Tensor Stokes formulations could be applied to integral viscoelastic formulations. However, numerical treatment of integral viscoelastic formulations is highly challenging. By employing stress splitting in the integral formulation for pure polymer melts, it is possible to reintegrate the diffusive operator into the momentum equation, akin to differential viscoelastic models. The EVSS and Tensor Stokes integral viscoelastic formulations for the Wagner integral model are described, with the goal of reducing the problem size from a three-field formulation to two-field formulation, where the stress tensor can be computed in a post-processing manner.

The numerical simulations of the three field differential viscoelastic models for zero solvent viscosity are the focus of this thesis. To achieve a highly accurate solution, the equations are discretized in space using Q_2, P_1^{disc}, Q_3 finite element triplet for the velocity, pressure and stress, respectively. The monolithic and decoupled approaches are employed to solve the three field EVSS and Tensor Stokes viscoelastic formulations. For the simplest Oldroyd-B model, the formulations are tested on the square domain with a uniform and perturbed mesh. Error computations are performed between the analytic and numerical results for increasing values of the relaxation parameter λ where a good agreement between the analytic and numerical results is achieved.

The considered formulations are tested on a more complicated geometry i.e., four to one contraction for different viscoelastic fluids of interest that includes Oldroyd-B, Giesekus and PTT-exponential fluid. Convergence behavior for these fluid models on the curved contraction are studied with and without the application of edge oriented (EO) stabilization in the momentum and as well as constitutive equation for stress. It was observed that the application of EO stabilization in the momentum equation or the constitutive equation on one hand helps to achieve faster convergence for lower relaxation parameter λ and on the other hand helps to obtain results at high values of the relaxation parameter. A comparison of computational costs between the decoupled and monolithic formulations is conducted, revealing that the decoupled formulation outperforms the monolithic formulation.

As the relaxation parameter λ is increased, the viscoelastic effects increase that means fluid begins to behave more like solid as compared to liquid which in turn arises the numerical difficulties. Certain limits of the relaxation parameter λ are observed by employing both the decoupled and monolithic approach for all three viscoelastic models. We moves forward with the aim to obtain results at very high values of the relaxation parameter λ. For this purpose, the non-stationary decoupled EVSS formulation in the PTT exponential

model is considered. Here in the decoupled formulation, the Stokes part is solved using UMFPACK where FGMRES solver is employed to solve the stress equation. Also, instead of considering constant time step size an adaptive stepping technique is developed with the aim to obtain the results at very high values of the relaxation parameter λ. The use of adaptive time stepping technique helped to achieve the results for highly viscoelastic fluids without encountering any limits in the relaxation parameter λ. The simulations were intentionally stopped after $\lambda = 50$, implying that no lambda wall could be observed using this technique. The effectiveness of the adaptive time stepping technique motivated us to refine it and develop a modified adaptive time stepping technique. The goal was to strikes a balance between the computational cost associated with the FGMRES solver (in terms of maximum number of iterations) and the total number of steps required to achieve a steady-state solution. At the next step of the decoupled formulation, the UMFPACK solver is replaced by the multigrid for the Stokes flow. Numerical results were successfully obtained until a high relaxation parameter $\lambda = 5$ however, no convergence could be attained beyond that using multigrid solver.

Future work will focus on addressing the limitations of the Multigrid solver in the current setup. Furthermore, the developed adaptive time stepping routine is intended to find the limits of the relaxation parameter λ for complicated models like Giesekus model on complex geometries for example flow around the cylinder. Additionally, we aim to explore the potential benefits of the Tensor Stokes formulation, which could provide an explicit expression for tensor-based viscosity. This approach could be valuable for simulating complex viscoelastic fluids.

References

[1] P. Amestoy, A. Buttari, I. S. Duff, A. Guermouche, J. Y. L'Excellent, and B. Uçar. Multifrontal method. *Encyclopedia of Parallel Computing*, pages 1209–1216, 2011.

[2] R. C. Armstrong, A. N. Baris, R. A. Brown, and P. W. Yeh. Galerkin finite element analysis of complex viscoelastic flows. *Computer Methods in Applied Mechanics and Engineering*, 58:201–226, 1986.

[3] D. N. Arnold, D. Boffi, and R. S. Falk. Approximation by quadrilateral finite elements. *Mathematics of computation*, 71:909–922, 2002.

[4] F. P. T. Baaijens. An iterative solver for the DEVSS/DG method with application to smooth and non-smooth flows of the upper convected maxwell fluid. *Journal of Non-Newtonian Fluid Mechanics*, 75:119 – 138, 1998.

[5] F. P. T. Baaijens. Mixed finite element methods for viscoelastic flow analysis: a review. *Journal of Non-Newtonian Fluid Mechanics*, 79:361–385, 1998.

[6] F. P. T. Baaijens, S. H. A. Selen, H. P. W. Baaijens, and H. E. H. Meijer. Viscoelastic flow past a confined cylinder of a low density polyethylene melt. *Journal of Non-Newtonian Fluid Mechanics*, 68:173–203, 1997.

[7] J. Baranger and D. Sandri. A formulation of Stokes's problem and the linear elasticity equations suggested by the Oldroyd model for viscoelastic flow. *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique*, 2:331–345, 1992.

[8] K. J. Bathe. Computational fluid and solid mechanics. *Elsevier, 2001. Proceedings First MIT Conference on Computational Fluid and Solid Mechanics*, 2.

[9] A. Beris, R. Armstrong, and R. Brown. Finite element calculation of viscoelastic flow in a journal bearing: I. small eccentricities. *Journal Non-Newtonian Fluid Mechanics*, 16:141–172, 1984.

[10] D. Boffi and L. Gastaldi. On the quadrilateral Q2–P1 element for the Stokes problem. *International Journal for Numerical Methods in Fluids*, 39:1001–1011, 2002.

[11] D . V Boger and K Walters. *Rheological Phenomena in Focus*. 1993.

[12] P. Bollada and T. Phillips. A modified deformation field method for integral constitutive models. *Journal of Non-Newtonian Fluid Mechanics*, 163:78–87, 2009.

[13] A. Bonito and E. Burman. A continuous interior penalty method for viscoelastic flows. *SIAM Journal of Scientific Computing,*, 30:1156–1177, 2008.

[14] F. Brezzi and M. Fortin. Mixed and hybrid finite element methods. *Springer Series in Computational Mathematics*, 15, 1991.

[15] R. A. Brown, R. C. Armstrong, A. N. Baris, and P. W. Yeh. Galerkin finite element analysis of complex viscoelastic flows. *Computer Methods in Appied Mechanics and Engineering*, 58:201–226, 1986.

[16] M. J. Crochet. Numerical simulation of viscoelastic flow: A review. *Rubber Chemistry and Technology*, 62:426–455, 1989.

[17] H. Damanik, J. Hron, A. Ouazzi, and S. Turek. Monolithic Newton-multigrid solution techniques for incompressible nonlinear flow models. *Numerical Methods in Fluids*, 71:208–222, 2013.

[18] T. A. Davis. Algorithm 832: UMFPACK V4.3—an unsymmetric-pattern multifrontal method. *ACM Transactions on Mathematical Software*, 30:196–199, 2004.

[19] T. A. Davis and Y. Hu. The university of Florida sparse matrix collection. *ACM Transactions on Mathematical Software*, 38:1, 2011.

[20] H. Elman, D. Silvester, and A. Wathen. Finite elements and fast iterative solvers: With application in incompressible fluid dynamics. *Numerical Mathematics and Scientific Computing, Oxford University Press*, 2006.

[21] L. L. Ferras, J. M. Nobrega, and F. T. Pinho. Analytical solutions for channel flows of Phan- Thien–Tanner and Giesekus fluids under slip. *Journal of Non-Newtonian Fluid Mechanics*, 171-172:97–105, 2012.

[22] M. Fortin and A. Fortin. A new approach for the FEM simulation of viscoelastic flows. *Journal of Non-Newtonian Fluid Mechanics*, 32:295–310, 1989.

[23] M. Fortin, R. Guenette, and R. Pierre. Numerical analysis of the modified EVSS method. *Computer Methods in Applied Mechanics and Engineering*, 143:79–95, 1997.

[24] M. Fortin and R. Pierre. On the convergence of the mixed method of Crochet and Marchal for viscoelastic flows. *Computer Methods in Applied Mechanics and Engineering*, 73:341–350, 1989.

[25] L. P. Franca and R. Stenberg. Error analysis of galerkin least squares methods for the elasticity equations. *SIAM Journal of Numerical Analysis*, 28:1680–1697, 1991.

[26] H. Giesekus. A simple constitutive equation for polymer fluids based on the concept of deformation-dependent tensorial mobility. *Journal of Non-Newtonian Fluid Mechanics*, 11:69–109, 1982.

[27] V. Girault and P. A. Raviart. Finite element methods for Navier-Stokes equations. *Springer*, 1986.

[28] R. Guénette and M. Fortin. A new mixed finite element method for computing viscoelastic flows. *Journal of Non-Newtonian Fluid Mechanics*, 60:27–52, 1995.

[29] J. G. Heywood, R. Rannacher, and S Turek. Artificial boundaries and flux and pressure conditions for the incompressible Navier-Stokes equations. *International Journal for Numerical Methods in Fluids*, 22:325–352, 1992.

[30] J. Hron and S. Turek. A monolithic FEM/multigrid solver for ALE formulation of Fluid Structure interaction with application in Biomechanics. *Lecture Notes in Computational Science and Engineering*, 53:146–170, 2006.

[31] M. Hulsen, E. Peters, and B. Van den Brule. A new approach to the deformation fields method for solving complex flows using integral constitutive equations,. *Journal of Non-Newtonian Fluid Mechanics*, 98:201 – 221, 2001.

[32] M. A. Hulsen and P. D. Anderson. The deformation fields method revisited: Stable simulation of instationary viscoelastic fluid flow using integral models,. *Journal of Non-Newtonian Fluid Mechanics*, 262:68 – 78, 2018.

[33] M. A. Hulsen, R. Fattal, and R. Kupferman. Flow of viscoelastic fluids past a cylinder at high Weissenberg number: Stabilized simulations using matrix logarithms. *Journal of Non-Newtonian Fluid Mechanics*, 127:27–39, 2005.

[34] R. Keunings. On the high Weissenberg number problem. *Journal of Non-Newtonian Fluid Mechanics*, 20:209–226, 1986.

[35] R. Keunings. Finite element methods for integral viscoelastic fluids. 2003.

[36] J. Kroll and S. Turek. Evaluation of nonlinear differential models for the simulation of polymer melts. *Kautschuk Gummi Kunststoffe*, 317:48–52, 2017.

[37] R. G. Larson. Constitutive equations for polymer melts and solutions,. *Butterworths Series in Chemical Engineering, Butterworth-Heinemann*, 1988.

[38] W. J. Layton. Introduction to the numerical analysis of incompressible viscous flows. *SIAM*, 2008.

[39] D. L. Logan. A first course in the finite element method,. *University of Wisconsin–Platteville*.

[40] C. W. Macosko. Rheology: principles, measurements, and applications. *Wiley*, 1993.

[41] C. W. Macosko and R. G. Larson. Rheology : principles, measurements, and applications,. *New York*, 1994.

[42] J. Marchal and M. Crochet. A new mixed finite element for calculating viscoelastic flow. *Journal of Non-Newtonian Fluid Mechanics*, 26:77–114, 1987.

[43] M. A. Mendelson, P. W. Yeh, R. A. Brown, and R. C. Armstrong. Approximation error in finite element calculation of viscoelastic fluid flows. *Journal Non-Newtonian Fluid Mechanics*, 10:31–54, 1982.

[44] J. G. Oldroyd. On the formulation of rheological equations of state. *Proceedings of the Royal society a Mathematical, Physical and Engineering sciences*, 200:523–541, 1950.

[45] A. Ouazzi. Finite element simulation of nonlinear fluids: Application to granular material and powder. *Industrial and applied mathematics*, 2005.

[46] A. Ouazzi and S. Turek. Numerical methods and simulation techniques for flow with shear and pressure dependent viscosity. *Numerical Mathematics and Advanced Applications, Springer*, page 668–676, 2003.

[47] A. Ouazzi and S. Turek. Efficient multigrid and data structures for edge-oriented FEM stabilization. *Numerical Mathematics and Advanced Applications Enumath*, page 520–527, 2005.

[48] M. Perera and K. Walters. Long- range memory effects in flows involving abrupt changes in geometry, Part i. Flows associated with L-shaped and T-shaped geometries. *Journal of Non-Newtonian Fluid Mechanics*, 2:49–81, 1977.

[49] E. A .J .F. Peters, M. A. Hulsen, and B. H. A. A. van den Brule. Instationary Eulerian viscoelastic flow simulations using time separable Rivlin–Sawyers constitutive equations. *Journal of Non-Newtonian Fluid Mechanics*, 89:209–228, 2000.

[50] N. Phan-Thien. A nonlinear network viscoelastic model. *Journal of Rheology*, 22:259–283, 1978.

[51] N. Phan-Thien and R. I. Tanner. A new constitutive equation derived from network theory. *Journal of Non-Newtonian Fluid Mechanics*, 2:353–365, 1977.

[52] M. Picasso, J. Bonvin, and R. Stenberg. GLS and EVSS methods for a three-field Stokes problem arising from viscoelastic flows. *Computer Methods in Applied Mechanics and Engineering*, 29:3893–3914, 2001.

[53] D. Rajagopalan, R. C. Armstrong, and R. A. Brown. Finite element methods for calculation of steady, viscoelastic flow using constitutive equations with a Newtonian viscosity. *Journal of Non-Newtonian Fluid Mechanics*, 36:159–192, 1990.

[54] R. Rannacher and S. Turek. Simple nonconforming quadrilateral Stokes element. *Numerical Methods for Partial Differential Equations*, 8:97–111, 1992.

[55] V. Ruas. Finite element methods for the three-field Stokes system in $\mathbf{R}^3$:Galerkin methods. *ESAIM: Mathematical Modelling and Numerical Analysis*, 4:489–525, 1996.

[56] Y. Saad. Overview of Krylov subspace methods with applications to control problems. *Research Institute for Advanced Computer Science NASA Ames Research Center*, 1989.

[57] Y. Saad. Iterative methods for sparse linear systems. *2nd. SIAM*, 2003.

[58] R. Temam. Chapter I - the steady-state Stokes equations. *Studies in Mathematics and Its Applications, Elsevier*, 2:1–156, 1977.

[59] M. F. Tome. A finite difference technique for solving a time strain separable K-BKZ constitutive equation for two-dimensional moving free surface flows. *J. Comp. Phy.*, 311:114–141, 2016.

[60] M. F. Tome, J. Bertoco, C. M. Oishi, M. S. B. Araujo, D. Cruz, F. T. Pinho, and M. Vynnycky. A finite difference technique for solving a time strain separable K-

BKZ constitutive equation for two-dimensional moving free surface flows,. *Journal of Computational Physics*, 311:114 – 141, 2016.

[61] S. Turek and A. Ouazzi. Unified edge–oriented stabilization of nonconforming FEM for incompressible flow problems: Numerical investigations. *Journal of Numerical Mathematics*, 15:299–322, 2007.

[62] S. Turek, A. Ouazzi, and R. Schmachtel. Multigrid methods for stabilized nonconforming finite elements for incompressible flow involving the deformation tensor formulation. *Journal of Non-Newtonian Fluid Mechanics*, 10:235–248, 2002.

[63] S. Turek and M. Schafer. Benchmark computations of laminar flow around cylinder, flow simulation with high-performance computers II. *Notes on Numerical Fluid Mechanics*, 52:547–566, 1996.

[64] S. Vanka. Block-implicit multigrid solution of Navier-Stokes equations in primitive variables. *Journal of Computational Physics*, 65:138 – 158, 1986.

[65] R. Verfurth. A review of a posteriori error estimation and adaptive mesh refinement techniques. *Wiley and Teubner*, 1996.

[66] P. Westervoss. *The Tensor Diffusion approach as a novel technique for simulating viscoelastic fluid flows*. PhD Thesis, Faculty of Applied Mathematics and Numerics, Technical University of Dortmund, 2021.

[67] H. Wobker and S. Turek. Numerical studies of Vanka-type smoothers in computational solid mechanics. *Advances in Applied Mathematics and Mechanics*, 1:29–55, 2009.

[68] J. V. D. Zanden and M. Hulsen. Mathematical and physical requirements for successful computations with viscoelastic fluid models. *Journal of Non-Newtonian Fluid Mechanics*, 29:93 – 117, 1988.

A

Appendix

A.1 Component wise form of the viscoelastic formulations

The velocity field $\mathbf{u}$ has two components (u_1, u_2). The deformation tensor is given as:

$$\begin{aligned}\mathbf{D}(\mathbf{u}) &= \frac{1}{2}(\nabla\mathbf{u} + \nabla\mathbf{u}^T) \\ &= \frac{1}{2}\begin{pmatrix} 2\frac{\partial u_1}{\partial x} & \frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x} \\ \frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x} & 2\frac{\partial u_2}{\partial y} \end{pmatrix}\end{aligned} \tag{A.1}$$

The deformation $\mathbf{D}(\mathbf{u})$ is symmetric in nature.

A.1.1 Extra-stress tensor formulation

The component wise notation of the extra stress tensor formulation (2.4) is presented in this section. The stress tensor $\boldsymbol{S}$ is symmetric i.e., $(S_{12} = S_{21})$ and has the following form:

$$\boldsymbol{S} = \begin{pmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{pmatrix} \tag{A.2}$$

Momentum equation: Momentum equation in component wise notation is given as:

$$\begin{aligned}\frac{\partial u_1}{\partial t} - \eta_s\left(2\frac{\partial^2 u_1}{\partial x^2} + \frac{\partial^2 u_1}{\partial y^2} + \frac{\partial^2 u_2}{\partial x \partial y}\right) - \left(\frac{\partial S_{11}}{\partial x} + \frac{\partial S_{12}}{\partial y}\right) + \frac{\partial p}{\partial x} = 0 \\ \frac{\partial u_2}{\partial t} - \eta_s\left(\frac{\partial^2 u_1}{\partial x \partial y} + \frac{\partial^2 u_2}{\partial x^2} + 2\frac{\partial^2 u_2}{\partial y^2}\right) - \left(\frac{S_{21}}{\partial x} + \frac{\partial S_{22}}{\partial y}\right) + \frac{\partial p}{\partial y} = 0\end{aligned} \tag{A.3}$$

Continuity equation: Continuity equation in component wise notation is given as:

$$\frac{\partial u_1}{\partial x} + \frac{\partial u_2}{\partial y} = 0 \tag{A.4}$$

Constitutive equation: Let us first consider the upper convected term $\lambda \overset{\nabla}{\boldsymbol{S}}$ of the constitutive equation,

$$\overset{\nabla}{\boldsymbol{S}} = \frac{\partial \boldsymbol{S}}{\partial t} + (\mathbf{u} \cdot \nabla)\, \boldsymbol{S} - \nabla \mathbf{u}^T \cdot \boldsymbol{S} - \boldsymbol{S} \cdot \nabla \mathbf{u} \tag{A.5}$$

The component wise notation of the above equation is given as:

$$\begin{aligned}
A &= \frac{\partial S_{11}}{\partial t} + \left(u_1 \frac{\partial}{\partial x} + u_2 \frac{\partial}{\partial y} \right) S_{11} - \left(2S_{11} \frac{\partial u_1}{\partial x} + 2S_{12} \frac{\partial u_1}{\partial y} \right) \\
B &= \frac{\partial S_{12}}{\partial t} + \left(u_1 \frac{\partial}{\partial x} + u_2 \frac{\partial}{\partial y} \right) S_{12} - \left(S_{11} \frac{\partial u_2}{\partial x} + S_{12} \frac{\partial u_1}{\partial x} + S_{22} \frac{\partial u_1}{\partial y} + S_{12} \frac{\partial u_2}{\partial y} \right) \\
C &= \frac{\partial S_{22}}{\partial t} + \left(u_1 \frac{\partial}{\partial x} + u_2 \frac{\partial}{\partial y} \right) S_{22} - \left(2S_{12} \frac{\partial u_2}{\partial x} + 2S_{22} \frac{\partial u_2}{\partial y} \right)
\end{aligned} \tag{A.6}$$

The constitutive equation for the three viscoelastic models is given as:

1. Oldroyd-B model: The Oldroyd-B model (2.6) in component wise notation is given as:

$$\begin{aligned}
S_{11} + \lambda A &= 2\eta_p \frac{\partial u_1}{\partial x} \\
S_{12} + \lambda B &= \eta_p \left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x} \right) \\
S_{22} + \lambda C &= 2\eta_p \frac{\partial u_2}{\partial y}
\end{aligned} \tag{A.7}$$

2. Giesekus model: The Giesekus model (2.7) in component wise notation is given as:

$$\begin{aligned}
S_{11} + \frac{\alpha\lambda}{\eta_p}(S_{11}^2 + S_{12}^2) + \lambda A &= 2\eta_p \frac{\partial u_1}{\partial x} \\
S_{12} + \frac{\alpha\lambda}{\eta_p}(S_{11}S_{12} + S_{12}S_{22}) + \lambda B &= \eta_p \left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x} \right) \\
S_{22} + \frac{\alpha\lambda}{\eta_p}(S_{12}^2 + S_{22}^2) + \lambda C &= 2\eta_p \frac{\partial u_2}{\partial y}
\end{aligned} \tag{A.8}$$

3. PTT exponential model: The PTT exponential model (2.10) in component wise notation is given as:

$$\begin{aligned}
exp\left(\frac{\lambda\kappa}{\eta_p} \mathrm{Tr}(\boldsymbol{S}) \right) S_{11} + \lambda A &= 2\eta_p \frac{\partial u_1}{\partial x} \\
exp\left(\frac{\lambda\kappa}{\eta_p} \mathrm{Tr}(\boldsymbol{S}) \right) S_{12} + \lambda B &= \eta_p \left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x} \right) \\
exp\left(\frac{\lambda\kappa}{\eta_p} \mathrm{Tr}(\boldsymbol{S}) \right) S_{22} + \lambda C &= 2\eta_p \frac{\partial u_2}{\partial y}
\end{aligned} \tag{A.9}$$

   where $\mathrm{Tr}(\boldsymbol{S}) = S_{11} + S_{22}$.

Let us write the component wise notation of the elastic viscous stress splitting formulation.

A.1.2 Elastic viscous stress splitting formulation

The component wise notation of the elastic viscous stress splitting (2.12) is presented in this section. The stress tensor $\boldsymbol{E}$ is symmetric i.e., $(E_{12} = E_{21})$ and has the following form:

$$\boldsymbol{E} = \begin{pmatrix} E_{11} & E_{12} \\ E_{21} & E_{22} \end{pmatrix} \tag{A.10}$$

Momentum equation: Momentum equation in component wise notation is given as:

$$\begin{aligned} \frac{\partial u_1}{\partial t} - \eta_0 \left(2\frac{\partial^2 u_1}{\partial x^2} + \frac{\partial^2 u_1}{\partial y^2} + \frac{\partial^2 u_2}{\partial x \partial y} \right) - \left(\frac{\partial E_{11}}{\partial x} + \frac{\partial E_{12}}{\partial y} \right) + \frac{\partial p}{\partial x} = 0 \\ \frac{\partial u_2}{\partial t} - \eta_0 \left(\frac{\partial^2 u_1}{\partial x \partial y} + \frac{\partial^2 u_2}{\partial x^2} + 2\frac{\partial^2 u_2}{\partial y^2} \right) - \left(\frac{E_{21}}{\partial x} + \frac{\partial E_{22}}{\partial y} \right) + \frac{\partial p}{\partial y} = 0 \end{aligned} \tag{A.11}$$

Constitutive equation: Let us first consider the upper convected terms $\lambda \overset{\nabla}{\boldsymbol{E}}$ and $2\lambda\eta_p \mathbf{D}(\overset{\nabla}{\mathbf{u}})$ of the constitutive equation. The component wise notation of $\lambda \overset{\nabla}{\boldsymbol{E}}$ is given as:

$$\begin{aligned} F &= \frac{\partial E_{11}}{\partial t} + \left(u_1 \frac{\partial}{\partial x} + u_2 \frac{\partial}{\partial y} \right) E_{11} - \left(2E_{11}\frac{\partial u_1}{\partial x} + 2E_{12}\frac{\partial u_1}{\partial y} \right) \\ G &= \frac{\partial E_{12}}{\partial t} + \left(u_1 \frac{\partial}{\partial x} + u_2 \frac{\partial}{\partial y} \right) E_{12} - \left(E_{11}\frac{\partial u_2}{\partial x} + E_{12}\frac{\partial u_1}{\partial x} + E_{22}\frac{\partial u_1}{\partial y} + E_{12}\frac{\partial u_2}{\partial y} \right) \\ H &= \frac{\partial E_{22}}{\partial t} + \left(u_1 \frac{\partial}{\partial x} + u_2 \frac{\partial}{\partial y} \right) E_{22} - \left(2E_{12}\frac{\partial u_2}{\partial x} + 2E_{22}\frac{\partial u_2}{\partial y} \right) \end{aligned} \tag{A.12}$$

The component wise notation of $2\lambda\eta_p \mathbf{D}(\overset{\nabla}{\mathbf{u}})$ is given as:

$$\begin{aligned} I &= \frac{\partial D_{11}}{\partial t} + \left(u_1 \frac{\partial}{\partial x} + u_2 \frac{\partial}{\partial y} \right) D_{11} - \left(2D_{11}\frac{\partial u_1}{\partial x} + 2D_{12}\frac{\partial u_1}{\partial y} \right) \\ J &= \frac{\partial D_{12}}{\partial t} + \left(u_1 \frac{\partial}{\partial x} + u_2 \frac{\partial}{\partial y} \right) D_{12} - \left(D_{11}\frac{\partial u_2}{\partial x} + D_{12}\frac{\partial u_2}{\partial y} + D_{12}\frac{\partial u_1}{\partial x} + D_{22}\frac{\partial u_1}{\partial y} \right) \\ K &= \frac{\partial D_{22}}{\partial t} + \left(u_1 \frac{\partial}{\partial x} + u_2 \frac{\partial}{\partial y} \right) D_{22} - \left(2D_{22}\frac{\partial u_2}{\partial y} + 2D_{12}\frac{\partial u_2}{\partial x} \right) \end{aligned} \tag{A.13}$$

The constitutive equation for the three viscoelastic models is given as:

1. The Oldroyd-B model (2.15) in component wise notation is given as:

$$\begin{aligned} E_{11} + \lambda F &= -2\eta_p \lambda I \\ E_{12} + \lambda G &= -2\eta_p \lambda J \\ E_{22} + \lambda H &= -2\eta_p \lambda K \end{aligned} \tag{A.14}$$

2. Giesekus model: The Giesekus model (2.16) in component wise notation is given as:

$$\begin{aligned} E_{11} + 4\alpha\lambda\eta_p\left(D_{11}^2 + D_{12}^2\right) + 2\alpha\lambda\left(2D_{11}E_{11} + 2D_{12}E_{12}\right) \\ +\frac{\alpha\lambda}{\eta_p}\left(E_{11}^2 + E_{12}^2\right) + \lambda F = -2\eta_p\lambda I \\ E_{12} + 4\alpha\lambda\eta_p\left(D_{11}D_{12} + D_{12}D_{22}\right) + \frac{\alpha\lambda}{\eta_p}\left(E_{11}E_{12} + E_{22}E_{12}\right) \\ +2\alpha\lambda\left(D_{11}E_{12} + D_{12}E_{22} + D_{12}E_{11} + D_{22}E_{12}\right) + \lambda G = -2\eta_p\lambda J \\ E_{22} + 4\alpha\lambda\eta_p\left(D_{22}^2 + D_{12}^2\right) + 2\alpha\lambda\left(2D_{22}E_{22} + 2D_{12}E_{12}\right) \\ +\frac{\alpha\lambda}{\eta_p}\left(E_{22}^2 + E_{12}^2\right) + \lambda H = -2\eta_p\lambda K \end{aligned} \tag{A.15}$$

3. PTT exponential model: The PTT exponential model (2.19) in component wise notation is given as:

$$\begin{aligned} exp\left(\frac{\lambda\kappa}{\eta_p}\mathrm{Tr}\left(2\eta_p\mathbf{D}(\mathbf{u}) + \boldsymbol{E}\right)\right)\left(E_{11} + 2\eta_p\frac{\partial u_1}{\partial x}\right) + \lambda F = 2\eta_p\frac{\partial u_1}{\partial x} - 2\eta_p\lambda I \\ exp\left(\frac{\lambda\kappa}{\eta_p}\mathrm{Tr}\left(2\eta_p\mathbf{D}(\mathbf{u}) + \boldsymbol{E}\right)\right)\left(E_{12} + \eta_p\left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x}\right)\right) + \lambda G = \\ \eta_p\left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x}\right) - 2\eta_p\lambda J \\ exp\left(\frac{\lambda\kappa}{\eta_p}\mathrm{Tr}\left(2\eta_p\mathbf{D}(\mathbf{u}) + \boldsymbol{E}\right)\right)\left(E_{22} + 2\eta_p\frac{\partial u_2}{\partial y}\right) + \lambda H = 2\eta_p\frac{\partial u_2}{\partial y} - 2\eta_p\lambda K \end{aligned} \tag{A.16}$$

where $\mathrm{Tr}\left(2\eta_p\mathbf{D}(\mathbf{u}) + \boldsymbol{E}\right) = E_{11} + E_{22} + 2\eta_p\left(\frac{\partial u_1}{\partial x} + \frac{\partial u_2}{\partial y}\right)$

Let us write the component wise notation of the Tensor Stokes formulation.

A.1.3 Tensor Stokes formulation

The component wise notation of the Tensor Stokes formulation (2.22) is presented in this section. The stress tensor $\boldsymbol{T}$ has the following components:

$$\boldsymbol{T} = \begin{pmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{pmatrix} \tag{A.17}$$

Momentum equation: Momentum equation in component wise notation is given as:

$$\rho\frac{\partial u_1}{\partial t} - \frac{1}{2}\left(\frac{\partial}{\partial x}\left(2T_{11}D_{11} + 2T_{12}D_{12}\right) + \frac{\partial}{\partial y}\left(T_{11}D_{12} + T_{12}D_{22} + T_{21}D_{11} + T_{22}D_{12}\right)\right) + \frac{\partial p}{\partial x} = 0$$

$$\rho\frac{\partial u_2}{\partial t} - \frac{1}{2}\left(\frac{\partial}{\partial x}\left(T_{21}D_{11} + T_{22}D_{12} + T_{11}D_{12} + T_{12}D_{22}\right) + \frac{\partial}{\partial y}\left(2T_{21}D_{12} + 2T_{22}D_{22}\right)\right) + \frac{\partial p}{\partial y} = 0 \tag{A.18}$$

Constitutive equation: Let us first consider the upper convected term $(\boldsymbol{T} \cdot \overset{\nabla}{\mathbf{D}})$ of the constitutive equation (2.23). The component wise notation is given as:

$$\begin{aligned}
L = \frac{\partial}{\partial t}\left(T_{11}D_{11} + T_{12}D_{12}\right) + \left(u_1\frac{\partial}{\partial x} + u_2\frac{\partial}{\partial y}\right)\left(T_{11}D_{11} + T_{12}D_{12}\right) \\
-\lambda\Big(2\frac{\partial u_1}{\partial x}\left(T_{11}D_{11} + T_{12}D_{12}\right) \\
+\frac{\partial u_1}{\partial y}\left(T_{21}D_{11} + T_{22}D_{21} + T_{11}D_{12} + T_{12}D_{22}\right)\Big) \\
M = \frac{\partial}{\partial t}\left(T_{11}D_{12} + T_{12}D_{22}\right) + \left(u_1\frac{\partial}{\partial x} + u_2\frac{\partial}{\partial y}\right)\left(T_{11}D_{12} + T_{12}D_{22}\right) \\
-\lambda\Big(\frac{\partial u_1}{\partial x}\left(T_{11}D_{12} + T_{12}D_{22}\right) + \frac{\partial u_1}{\partial y}\left(T_{21}D_{12} + T_{22}D_{22}\right) \\
+\frac{\partial u_2}{\partial x}\left(T_{11}D_{11} + T_{12}D_{12}\right) + \frac{\partial u_2}{\partial y}\left(T_{11}D_{12} + T_{12}D_{22}\right)\Big) \\
N = \frac{\partial}{\partial t}\left(T_{21}D_{11} + T_{22}D_{21}\right) + \left(u_1\frac{\partial}{\partial x} + u_2\frac{\partial}{\partial y}\right)\left(T_{21}D_{11} + T_{22}D_{21}\right) \\
-\lambda\Big(\frac{\partial u_1}{\partial x}\left(T_{21}D_{11} + T_{22}D_{21}\right) + \frac{\partial u_1}{\partial y}\left(T_{21}D_{12} + T_{22}D_{22}\right) \\
+\frac{\partial u_2}{\partial x}\left(T_{11}D_{11} + T_{12}D_{12}\right) + \frac{\partial u_2}{\partial y}\left(T_{21}D_{11} + T_{22}D_{21}\right)\Big) \\
O = \frac{\partial}{\partial t}\left(T_{21}D_{12} + T_{22}D_{22}\right) + \left(u_1\frac{\partial}{\partial x} + u_2\frac{\partial}{\partial y}\right)\left(T_{22}D_{22} + T_{21}D_{12}\right) \\
-\lambda\Big(\frac{\partial u_2}{\partial x}\left(T_{21}D_{11} + T_{22}D_{12} + T_{11}D_{12} + T_{12}D_{22}\right) \\
+\frac{\partial u_2}{\partial y}\left(2T_{21}D_{12} + 2T_{22}D_{22}\right)\Big)
\end{aligned} \tag{A.19}$$

The constitutive equation for the three viscoelastic models is given as:

1. Oldroyd-B model: The Oldroyd-B model (2.24) in component wise notation is given as:

$$\begin{aligned}
T_{11}D_{11} + T_{12}D_{12} + \lambda L &= 2\eta_p \frac{\partial u_1}{\partial x} \\
T_{11}D_{12} + T_{12}D_{22} + \lambda M &= \eta_p \left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x} \right) \\
T_{21}D_{11} + T_{22}D_{21} + \lambda N &= \eta_p \left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x} \right) \\
T_{21}D_{12} + T_{22}D_{22} + \lambda O &= 2\eta_p \frac{\partial u_2}{\partial y}
\end{aligned} \tag{A.20}$$

2. Giesekus model: The Giesekus model (2.25) is obtained the addition of the quadratic term $\frac{\alpha\lambda}{\eta_p}\left((\boldsymbol{T}\cdot\mathbf{D}(\mathbf{u}))\cdot((\boldsymbol{T}\cdot\mathbf{D}(\mathbf{u}))\right)$ to the Oldroyd-B model. Let us first write the quadratic term in component wise notation in the following way:

$$\begin{aligned}
P = T_{11}^2 D_{11}^2 + T_{12}^2 D_{12}^2 + 2D_{11}D_{12}T_{11}T_{12} + D_{11}D_{12}T_{11}T_{21} \\
+D_{12}^2 T_{11}T_{22} + D_{11}D_{22}T_{12}T_{21} + D_{22}D_{12}T_{12}T_{22} \\
Q = T_{11}^2 D_{11}D_{12} + D_{11}D_{22}T_{11}T_{12} + D_{12}^2 T_{11}T_{12} + D_{12}D_{22}{T_{12}}^2 + \\
D_{12}^2 T_{11}T_{21} + D_{12}D_{22}T_{11}T_{22} + D_{12}D_{22}T_{12}T_{21} + D_{22}^2 T_{12}T_{22} \\
R = D_{11}^2 T_{11}T_{21} + D_{11}D_{12}T_{12}T_{21} + D_{11}D_{12}T_{11}T_{22} + D_{12}^2 T_{12}T_{22} + \\
D_{11}D_{12}T_{21}^2 + D_{12}^2 T_{21}T_{22} + D_{11}D_{22}T_{22}T_{21} + D_{22}D_{21}T_{22}^2 \\
S = T_{21}^2 D_{12}^2 + T_{22}^2 D_{22}^2 + 2D_{22}D_{12}T_{21}T_{22} + D_{11}D_{12}T_{11}T_{21} \\
+D_{12}^2 T_{11}T_{22} + D_{11}D_{22}T_{12}T_{21} + D_{22}D_{12}T_{12}T_{22}
\end{aligned} \tag{A.21}$$

The component wise notation of the Giesekus model is given as:

$$\begin{aligned}
T_{11}D_{11} + T_{12}D_{12} + \frac{\alpha\lambda}{\eta_p}P + \lambda L &= 2\eta_p \frac{\partial u_1}{\partial x} \\
T_{11}D_{12} + T_{12}D_{22} + \frac{\alpha\lambda}{\eta_p}Q + \lambda M &= \eta_p \left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x} \right) \\
T_{21}D_{11} + T_{22}D_{21} + \frac{\alpha\lambda}{\eta_p}R + \lambda N &= \eta_p \left(\frac{\partial u_1}{\partial y} + \frac{\partial u_2}{\partial x} \right) \\
T_{21}D_{12} + T_{22}D_{22} + \frac{\alpha\lambda}{\eta_p}S + \lambda O &= 2\eta_p \frac{\partial u_2}{\partial y}
\end{aligned} \tag{A.22}$$

3. PTT exponential model: The PTT exponential model (2.28) in component wise notation is given as:

$$
\begin{aligned}
exp\left(\frac{\lambda\kappa}{\eta_p}\mathrm{Tr}\left(\boldsymbol{T}\cdot\mathbf{D}(\mathbf{u})\right)\right)\left(T_{11}D_{11}+T_{12}D_{12}\right)+\lambda L &= 2\eta_p\frac{\partial u_1}{\partial x}\\
exp\left(\frac{\lambda\kappa}{\eta_p}\mathrm{Tr}\left(\boldsymbol{T}\cdot\mathbf{D}(\mathbf{u})\right)\right)\left(T_{11}D_{12}+T_{12}D_{22}\right)+\lambda M &= \eta_p\left(\frac{\partial u_1}{\partial y}+\frac{\partial u_2}{\partial x}\right)\\
exp\left(\frac{\lambda\kappa}{\eta_p}\mathrm{Tr}\left(\boldsymbol{T}\cdot\mathbf{D}(\mathbf{u})\right)\right)\left(T_{21}D_{11}+T_{22}D_{21}\right)+\lambda N &= \eta_p\left(\frac{\partial u_1}{\partial y}+\frac{\partial u_2}{\partial x}\right)\\
exp\left(\frac{\lambda\kappa}{\eta_p}\mathrm{Tr}\left(\boldsymbol{T}\cdot\mathbf{D}(\mathbf{u})\right)\right)\left(T_{21}D_{12}+T_{22}D_{22}\right)+\lambda O &= 2\eta_p\frac{\partial u_2}{\partial y}
\end{aligned}
\tag{A.23}
$$

where $\mathrm{Tr}\left(\boldsymbol{T}\cdot\mathbf{D}(\mathbf{u})\right)=T_{11}D_{11}+T_{12}D_{12}+T_{21}D_{12}+T_{22}D_{22}$

A.2 The operators in the matrix-vector notation

The operators employed in the above matrix structure (4.14) are given as follows:

$$
\begin{aligned}
A &= \rho\int_\Omega\frac{\partial\mathbf{u}}{\partial t}\boldsymbol{v}\,dx+2\eta_0\int_\Omega\mathbf{D}(\mathbf{u}):\mathbf{D}(\boldsymbol{v})\,dx\\
B &= -\int_\Omega p\nabla\cdot\boldsymbol{v}\,dx\\
R &= \int_\Omega\boldsymbol{E}:\mathbf{D}\left(\boldsymbol{v}\right)\,dx\\
B^T &= \int_\Omega q\,\nabla\cdot\mathbf{u}\,dx
\end{aligned}
\tag{A.24}
$$

In order to define the operator $K_{\overset{\nabla}{\mathbf{D}(\mathbf{u})}}$ and Z, the function $h\left(2\eta_p\mathbf{D}\left(\mathbf{u}\right)+\mathbf{E}\right)$ requires special attention. The function h=0 for Oldroyd-B and PTT-exponential model, so it directly vanishes from the constitutive equation. However, for the Giesekus model function h is given as:

$$
h\left(2\eta_p\mathbf{D}\left(\mathbf{u}\right)+\boldsymbol{E}\right)=\frac{\alpha\lambda}{\eta_p}\left(\left(2\eta_p\mathbf{D}(\mathbf{u})+\boldsymbol{E}\right)\cdot\left(2\eta_p\mathbf{D}(\mathbf{u})+\boldsymbol{E}\right)\right)
$$

After performing the matrix multiplication, only one of the terms from the function above goes to the operator $K_{\overset{\nabla}{\mathbf{D}(\mathbf{u})}}$ i.e., $4\alpha\lambda\eta_p\left(\mathbf{D}(\mathbf{u})\cdot\mathbf{D}(\mathbf{u})\right)$ and the rest goes to operator Z.

$$
\begin{aligned}
K_{\overset{\nabla}{\mathbf{D}(\mathbf{u})}} = \int_\Omega f(2\eta_p\mathbf{D}\left(\mathbf{u}\right)+\boldsymbol{E},\lambda)\,2\eta_p\mathbf{D}\left(\mathbf{u}\right):\boldsymbol{\phi}\,dx+\int_\Omega\lambda 2\eta_p\overset{\nabla}{\mathbf{D}}(\mathbf{u}):\boldsymbol{\phi}\,dx+\\
\int_\Omega 4\alpha\lambda\eta_p\left(\mathbf{D}(\mathbf{u})\cdot\mathbf{D}(\mathbf{u})\right):\boldsymbol{\phi}\,dx-\int_\Omega 2\eta_p\mathbf{D}\left(\mathbf{u}\right):\boldsymbol{\phi}\,dx
\end{aligned}
\tag{A.25}
$$

$$Z = \int_{\Omega} f(2\eta_p \mathbf{D}(\mathbf{u}) + \mathbf{E}, \lambda)\, \boldsymbol{E} : \boldsymbol{\phi}\, dx + \lambda \int_{\Omega} \overset{\nabla}{\boldsymbol{E}} : \boldsymbol{\phi}\, dx + \int_{\Omega} \frac{\alpha\lambda}{\eta_p}(\boldsymbol{E} \cdot \boldsymbol{E}) + \int_{\Omega} 2\alpha\lambda(\mathbf{D}(\mathbf{u}) \cdot \boldsymbol{E} + \boldsymbol{E} \cdot \mathbf{D}(\mathbf{u})) : \boldsymbol{\phi}\, dx \tag{A.26}$$

The operators employed in the above matrix structure (4.15) are given as follows:

$$\begin{aligned} A &= \rho \int_{\Omega} \frac{\partial \mathbf{u}}{\partial t} \boldsymbol{v}\, dx + \frac{1}{2} \int_{\Omega} \left(\boldsymbol{T} \cdot \mathbf{D}(\mathbf{u}) + \mathbf{D}(\mathbf{u}) \cdot \boldsymbol{T}^T\right) : \mathbf{D}(\boldsymbol{v})\, dx \\ B &= -\int_{\Omega} p \nabla \cdot \boldsymbol{v}\, dx \\ B^T &= \int_{\Omega} q\, \nabla \cdot \mathbf{u}\, dx \\ K &= \int_{\Omega} -2\eta_p \mathbf{D}(\mathbf{u})\ : \boldsymbol{\phi}\, dx \\ M &= \lambda \int_{\Omega} (\boldsymbol{T} \cdot \overset{\nabla}{\mathbf{D}}(\mathbf{u})) : \boldsymbol{\phi}\, dx + \int_{\Omega} f(\boldsymbol{T} \cdot \mathbf{D}, \lambda)\, (\boldsymbol{T} \cdot \mathbf{D}) : \boldsymbol{\phi}\, dx + \int_{\Omega} h\, (\boldsymbol{T} \cdot \mathbf{D}) : \boldsymbol{\phi}\, dx \end{aligned} \tag{A.27}$$

With the definition of these operators, the coupled system of non-linear equations (4.14) and (4.15) are solved for both Elastic Viscous Stress Splitting and Tensor Stokes approaches, respectively.