Application of spectral elements to multi-mode viscoelastic flows

Verbeeten, W.M.H.

Award date:
1996

Application of spectral elements to multi-mode viscoelastic flows

W.M.H. Verbeeten

August 1996

WFW report 96.111

Supervisors: Ir. P.D. Anderson
Dr.Ir. F.N. van de Vosse

Professor: Prof.Dr.Ir. F.P.T. Baaijens

Eindhoven University of Technology
Faculty of Mechanical Engineering
Center for Polymers and Composites
Computational Mechanics

Summary

The spectral element method is analyzed for the simulation of viscoelastic polymeric fluids. The method combines the geometry flexibilities of the classical finite element technique and the high accuracy of spectral methods. In order to handle efficiently multiple relaxation modes, necessary for a realistic description of polymeric melts, a discontinuous discretization of the extra stress variable is applied. This allows an elimination of the variable on element level and thus reducing the computational time. A more stable numerical method is obtained by adding an extra variable and equation to the set of coupled non-linear equations: the Stabilized Discontinuous Galerkin method. The choices of the spaces, the numerical integration and the solution method is discussed in detail.

A simple Stokes Poisseuille flow shows an algebraic rate of convergence for augmentation of the elements. An exponential rate of convergence is detected by increasing the polynomial order. To test the numerical method, a benchmark problem of the steady flow past a falling sphere in a tube for a UCM constitutive model is examined. A good agreement with values from the literature is found. However, a loss of convergence occurs at lower Weissenberg numbers for higher polynomial degrees. An insufficient spatial resolution together with an ill-conditioning of the large system matrix causes this effect. The ill-conditioning becomes worse as the approximation degree is augmented.

Samenvatting

De toepassing van de Spectrale Elementen Methode voor de simulatie van visco-elastische polymere stromingen is onderzocht. Deze methode combineert de geometrische flexibiliteiten van de klassieke Eindige Elementen Methode met de hoge nauwkeurigheid van de Spectrale Methode. Om commerciele polymere smelten realistisch te kunnen beschrijven, zijn meerdere relaxatie modes noodzakelijk. Een discontinue benadering van de spanningen staat een eliminatie van deze spanningen op element niveau toe. Dit levert een efficiënte behandeling van de multiple relaxatie modes en reduceert de rekentijd. Een stabielere methode wordt verkregen door het toevoegen van een extra term en vergelijking aan de totale set van vergelijkingen die de stroming beschrijven. Zodoende wordt de Stabilized Discontinuous Galerkin methode verkregen. In dit verslag worden de keuzes van de numerieke oplosruimtes, integratie regel en oplosmethode uitvoerig behandeld.

Een simpele Poisseuille stroming door een buis onder Stokes' condities laat voor deze methode een algebraïsche convergentie zien als het aantal elementen toeneemt. Het ophogen van de orde van de benaderingspolynomen brengt een exponentiële convergentie teweeg. Het benchmark probleem van de stationaire stroming rond een vallend balletje in een buis is onderzocht om de numerieke methode te testen. De resultaten komen overeen met waarden uit de literatuur. Echter, er treedt verlies van convergentie op bij een lager Weissenberg getal als de orde van de polynoom wordt opgehoogd. Een onvoldoende fijne mesh samen met een slechte conditie van de grote systeem matrix veroorzaken dit effect: hoe hoger de orde, des te slechter de conditie van de matrix.

Contents

Notation

Quantities

A, a	scalar
$\vec{a}$	vector
$\boldsymbol{A}$	second order tensor
$\boldsymbol{I}$	second order unit tensor
$\mathcal{A}$	space
$\underline{A}$	matrix
$\underset{\sim}{a}$	vector

Operations and functions

$\boldsymbol{A}^c$	conjugation
$\vec{a} \cdot \vec{b}$, $\boldsymbol{A} \cdot \vec{b}$, $\vec{a} \cdot \boldsymbol{B}$, $\boldsymbol{A} \cdot \boldsymbol{B}$	inner product
$\boldsymbol{A} : \boldsymbol{B}$	double inner product
$\boldsymbol{A}^{-1}$	inversion
$tr(\boldsymbol{A})$	trace of a second order tensor
$\vec{\nabla}$	gradient operator
$\dot{a}$	material time dervative
$\underline{A}^T$	transponation of a matrix
$\frac{\partial a}{\partial t}$	spatial time dervative

Chapter 1

Introduction

Complex flows of viscoelastic fluids become increasingly more important in the modern industry, *e.g.* in plastics, polymers and food processing. Therefore, a lot of research has been performed on these flows, both on the rheological behaviour as well as on the numerical simulations. To further develop and optimize the processing techniques of viscoelastic flows (extrusion, multi-layer injection moulding, film-blowing, etc.), numerical simulations are of a practical relevance. The constitutive equation, which has to capture realistically the rheological behaviour, and the numerical problems, which increase when elasticity becomes dominant over viscous effects, generate the complexity of numerically solving viscoelastic flows.

Lately, the numerical calculations of the viscoelastic flows using mixed methods are mainly performed by two different approaches. The first is based on the classical low order methods. The finite element method is widely used in this approach; improvement of the numerical solutions is obtained by reformulations of the set of equations or introduction of artificial diffusivity or consistent upwinding.
Marchal and Crochet [1] introduced an $n \times n$-bilinear subdivision of the extra stress interpolation on a biquadratic velocity element (with n usually equal to four) to obtain a stable discretization scheme.
The Explicitly Elliptic Momentum Equation (EEME) method was developed to compute flows with UCM-like (Upper Convected Maxwell) differential models which have no solvent viscosity. Reformulation of the momentum equation leads to a second-order elliptic operator for which mixed finite element methods are appropriate (King *et al.* [2]).
A stable formulation is also obtained with the Elastic Viscous Stress Split (EVSS) formulation, where the constitutive equation contains a Newtonian solvent viscosity. An explicitly elliptic operator is introduced into the momentum equation by splitting the deviatoric stress into Newtonian and polymer contributions. This method was originally introduced by Mendelson *et al.* [3] and further developed by Rajagopalan *et al.* [4].
In the Streamline Upwind / Petrov Galerkin (SUPG) formulation of Brooks and Hughes [5] the hyperbolic character of the constitutive equation is taken into account.
These above mentioned methods all employ continuous interpolations of stress variables. A discontinuous approach of the stress was introduced by Lesaint and Raviart [6] and first applied for viscoelastic flows by Fortin and Fortin [7], to handle the advective parts of the constitutive equation. An advantage of this Discontinuous Galerkin (DG) method, is the more easily satisfied stress-velocity compatibility (see Ying [8] and Fortin and Pierre [9]). Later

on, Fortin *et al.* [10] applied the DG-formulation in combinations with EVSS. An efficient handling of multi-modes by applying the DG-method was shown by Baaijens [11].

More recently, as a second approach, Beris *et al.* [12] introduced the spectral techniques, based on high order polynomial degrees of the approximating functions, in the field of viscoelastic flow calculations. An advantage of using spectral methods is that a prescribed accuracy may be attained with considerable fewer degrees of freedom than alternative lower order methods. However, without the use of domain decomposition, it is practically impossible to apply spectral methods in general irregular domains. Spectral multi-domain methods have been successfully used in the viscoelastic context by Owens and Phillips [13], Souvaliotis and Beris [14] and Van Kemenade and Deville [15]. Talwar and Khomami [16] applied higher order finite element techniques (*hp*-type), also successfully.

The objective of this work is to investigate the performance of the Spectral Element Method (SEM) for the computation of *steady* viscoelastic flows. To reduce the number of degrees of freedom, with respect to the efficient handling of multi-modes, stresses will be eliminated at element level. The SEM is ideal for treating non-linearities and producing highly accurate solutions as long as singularities are not present to hamper their exponential rate of convergence (Canuto *et al.* [17]). Although the numerical program is fit to handle multi-modes, in this work only a single mode will be applied.

As a test problem, the steady flow past a falling sphere in a tube is examined. This is a benchmark problem, for it has a smooth geometry and is therefore suitable for numerical simulations and its experiment is easy to perform. However, the flow is much more complex than originally thought. It has been investigated in detail by, among others, Chilcott and Rallison [18], Harlen [19], Lunsmann *et al.* [20], and, more recently, with higher order methods by Fan Yurun and Crochet [21] and Owens [22].
As mentioned by Owens [22], the flow is extremely difficult to calculate for a number of reasons. It is characterized by thin stress boundary layers in the region near the sphere, combined with a strong elongational flow along the symmetry line downstream (Harlen [19]). An inadequate resolution of these fine stress features can give rise to huge discretization errors. This results in loss of convergence beyond a modest, maximum value of the dimensionless Weissenberg number We: the importance of elasticity in the flow, denoted with the same definition as the non-dimensional elasticity parameter of Crochet *et al.* [23]. Thus, a large amount of nodal points is needed near the surface of the sphere to obtain satisfying results, leading to significant computational time.

Where Lunsmann *et al.* [20] reported a failure in convergence of the falling sphere problem beyond $We \approx 1.6$ with the UCM-model, Fan Yurun and Crochet [21], with their higher order method, reached convergence up to $We = 2.0$. They used a EVSS-SUPG model to tackle this problem. Selen [24], using a stabilized DG-EVSS method and low order elements, even produced convergence at $We = 2.5$. As this last method also has the advantage of efficiently handling multiple modes, because of its elimination of the stress variables at element level, this model in combination with the Spectral Element Method is taken under investigation.

In the next chapter, the definition of the problem is formulated, followed by an outline of the computational method in Chapter 3. The Stabilized Discontinuous Galerkin method

will be discussed in detail. A short overview of the Spectral Element Method and numerical integration will be given. And the used solution method for the flow problem is pointed out. Chapter 4 will contain some numerical results and a discussion will be presented. First, a flow under Stokes' conditions is calculated to investigate the spatial convergence of the method. Next, to test the performance of the numerical method, the benchmark problem of the falling sphere in a tube is illustrated. Finally, in Chapter 5, some conclusions are drawn and recommendations are given.

Chapter 2

Problem definition

Isothermal flow of incompressible fluids in a two-dimensional domain Ω with boundary Γ is described by the equations for conservation of momentum and conservation of mass:

$$\varrho(\frac{\partial \vec{u}}{\partial t} + \vec{u} \cdot \vec{\nabla}\vec{u}) - \vec{\nabla} \cdot \boldsymbol{\sigma} = \vec{f} \qquad \text{in } \Omega, \tag{2.1}$$

$$\vec{\nabla} \cdot \vec{u} = 0 \qquad \text{in } \Omega, \tag{2.2}$$

where ϱ is the density, $\vec{u}$ denotes the velocity field, $\vec{\nabla}$ the gradient operator, $\boldsymbol{\sigma}$ the Cauchy stress tensor and $\vec{f}$ the body force per unit volume of fluid.
The Cauchy stress tensor $\boldsymbol{\sigma}$ is defined by:

$$\boldsymbol{\sigma} = -p\boldsymbol{I} + \boldsymbol{\tau}, \tag{2.3}$$

where p is a pressure term and $\boldsymbol{\tau}$ is the extra stress tensor, which has to be defined by a constitutive model.
The Dirichlet boundary and the initial conditions complete the problem:

$$\vec{u}(\vec{x}, t) = \vec{u}_\Gamma(\vec{x}, t) \qquad \text{on } \Gamma, \tag{2.4}$$

$$\vec{u}(\vec{x}, 0) = \vec{0} \qquad \text{in } \Omega, \tag{2.5}$$

$$p(\vec{0}, t) = 0 \qquad \text{in } \Omega, \tag{2.6}$$

where Γ is the boundary with boundary conditions on velocity. In the sequel of this report, these boundary conditions will complete all the problems. Also, the equations will hold over the domain Ω, unless indicated differently.

For rheological behaviour of the viscoelastic fluid, a multi-mode approach of the extra stress tensor is considered, defined by:

$$\boldsymbol{\tau} = 2\eta_s \boldsymbol{D}_u + \sum_{i=1}^{M} \boldsymbol{\tau}_i \ , \tag{2.7}$$

where η_s denotes the viscosity of the pure viscous mode (often called the solvent viscosity), $\boldsymbol{D}_u$ the rate of deformation tensor defined as $2\boldsymbol{D}_u = \left(\vec{\nabla}\vec{u}\right)^c + \left(\vec{\nabla}\vec{u}\right)$ and i is the index for each mode.

A general constitutive equation to model the fluid [25, 26] is:

$$\lambda_i \overset{\diamond}{\boldsymbol{\tau}}_i + \boldsymbol{Y}_i \cdot \boldsymbol{\tau}_i = 2\eta_i \boldsymbol{D}_u \ , \tag{2.8}$$

with $\overset{\diamond}{\boldsymbol{\tau}}$ representing the Gordon-Showalter derivative of the extra stress tensor:

$$\overset{\diamond}{\boldsymbol{\tau}}_i = \frac{\partial \boldsymbol{\tau}_i}{\partial t} + \vec{u} \cdot \vec{\nabla}\boldsymbol{\tau}_i - (\boldsymbol{L} - \xi_i \boldsymbol{D}_u) \cdot \boldsymbol{\tau}_i - \boldsymbol{\tau}_i \cdot (\boldsymbol{L} - \xi_i \boldsymbol{D}_u)^c \ . \tag{2.9}$$

Here, λ_i is the relaxation time, $\boldsymbol{L}$ is the velocity gradient defined by $\boldsymbol{L} = \left(\vec{\nabla}\vec{u}\right)^c$, ξ_i is a dimensionless material parameter and η_i is the kinematic viscosity of the mode. For each constitutive model a different tensor $\boldsymbol{Y}$ is used.
For some models the tensor $\boldsymbol{Y}$ is given below.

PTTa model (Phan-Thien-Tanner):

$$\boldsymbol{Y}_i = e^{\frac{\varepsilon_i \lambda_i}{\eta_i} tr(\boldsymbol{\tau}_i)} \boldsymbol{I} \qquad -1 < \xi_i < 1. \tag{2.10}$$

PTTb model:

$$\boldsymbol{Y}_i = (1 + \frac{\varepsilon_i \lambda_i}{\eta_i} tr(\boldsymbol{\tau}_i))\boldsymbol{I} \qquad -1 < \xi_i < 1. \tag{2.11}$$

Giesekus model:

$$\boldsymbol{Y}_i = (\boldsymbol{I} + \frac{\alpha_i \lambda_i}{\eta_i} \boldsymbol{\tau}_i) \qquad \xi_i = 0. \tag{2.12}$$

UCM model (Upper Convected Maxwell):

$$\boldsymbol{Y}_i = \boldsymbol{I} \qquad \xi_i = 0. \tag{2.13}$$

where ε_i and α_i are dimensionless parameters. Detailed information on the constitutive equations can be found in Bird [25] and Larson [26].
Defining an operator $\mathcal{L}^*$:

$$\mathcal{L}_i^* \boldsymbol{\tau}_i = \lambda_i \{\vec{u} \cdot \vec{\nabla}\boldsymbol{\tau}_i - (\boldsymbol{L} - \xi_i \boldsymbol{D}_u) \cdot \boldsymbol{\tau}_i - \boldsymbol{\tau}_i \cdot (\boldsymbol{L} - \xi_i \boldsymbol{D}_u)^c\} \ . \tag{2.14}$$

The constitutive equation and the equations for conservation of momentum and mass of the flow problem can now be written as:

$$\lambda_i \frac{\partial \boldsymbol{\tau}_i}{\partial t} + \mathcal{L}_i^* \boldsymbol{\tau}_i + \boldsymbol{Y}_i \cdot \boldsymbol{\tau}_i - 2\eta_i \boldsymbol{D}_u = \boldsymbol{0} \qquad i = 1, \ldots, M \ , \tag{2.15}$$

$$\varrho\left(\frac{\partial \vec{u}}{\partial t} + \vec{u}\cdot\vec{\nabla}\vec{u}\right) - \vec{\nabla}\cdot\left(-p\boldsymbol{I} + 2\eta_s\boldsymbol{D}_u + \sum_{i=1}^{M}\boldsymbol{\tau}_i\right) - \vec{f} = \vec{0}\ , \tag{2.16}$$

$$\vec{\nabla}\cdot\vec{u} = 0\ . \tag{2.17}$$

As is customary in the classical fluid dynamics, the above mentioned equations will be made dimensionless using a suitable characteristic velocity U, length L and time of the deformation process θ. The dimensionless set of equations then reduces to:

$$De_i\frac{\partial \check{\boldsymbol{\tau}}_i}{\partial \check{t}} + We_i\,\check{\mathcal{L}}_i^{*}\check{\boldsymbol{\tau}}_i + \boldsymbol{Y}_i\cdot\check{\boldsymbol{\tau}}_i - 2\check{\eta}_i\check{\boldsymbol{D}}_u = \boldsymbol{0} \qquad i = 1,\ldots,M\ , \tag{2.18}$$

$$Sr\frac{\partial \check{\vec{u}}}{\partial \check{t}} + \check{\vec{u}}\cdot\check{\vec{\nabla}}\check{\vec{u}} - \frac{1}{Re}\check{\vec{\nabla}}\cdot\left(-\check{p}\boldsymbol{I} + 2\check{\eta}_s\check{\boldsymbol{D}}_u + \sum_{i=1}^{M}\check{\boldsymbol{\tau}}_i\right) - \frac{1}{Fr}\check{\vec{f}} = \vec{0}\ , \tag{2.19}$$

$$\check{\vec{\nabla}}\cdot\check{\vec{u}} = 0\ , \tag{2.20}$$

- Deborah number: $De_i = \dfrac{\lambda_i}{\theta}$,
- Weissenberg number: $We_i = \dfrac{\lambda_i U}{L}$,
- Strouhal number: $Sr = \dfrac{L}{U\theta}$,
- Reynolds number: $Re = \dfrac{\varrho U L}{\eta}$ with $\eta = \eta_s + \sum_{i=1}^{M}\eta_i$,
- Froude number: $Fr = \dfrac{U^2}{gL}$,

where $\check{x}$ are the dimensionless physical variables. The Deborah number is defined as the ratio of a characteristic time of the mode to a characteristic time of the deformation process, the Weissenberg number is denoting the importance of elasticity of the mode in the flow. The Strouhal number is given by the ratio of the instationary inertia forces to the stationary inertia forces, the Reynolds number is defined as the stationary inertia forces divided by the viscous forces and the Froude number is specified as the stationary inertia forces in proportion to the external and gravitational forces. In the sequel, the $\check{x}$-notation is dropped for convenience of presentation.

In the sequel, the stationary inertia forces will be neglected because they are small compared to the viscous forces at conditions investigated in this work, $Re \ll 1$. As we are only interested in the steady state solution, the characteristic time of the deformation process is large $\theta \gg 1$ and therefore the Deborah and Strouhal numbers are very small $De \ll 1, Sr \ll 1$. Furthermore, only plane and axisymmetric flow situations without swirl will be considered at the absence of gravitational forces. The problem of the Eqn. (2.18) to (2.20) then reduces to:

Problem MP *Find* $(\boldsymbol{\tau}_i, \vec{u}, p, \bar{\boldsymbol{D}})$ *such that*

$$We_i \, \mathcal{L}_i^* \boldsymbol{\tau}_i + \boldsymbol{Y}_i \cdot \boldsymbol{\tau}_i - 2\eta_i \boldsymbol{D}_u = \boldsymbol{0} \qquad i = 1, \ldots, M \; , \tag{2.21}$$

$$-\frac{1}{Re} \vec{\nabla} \cdot \left(-p\boldsymbol{I} + 2\eta_s \boldsymbol{D}_u + \sum_{i=1}^{M} \boldsymbol{\tau}_i \right) = \vec{0} \; , \tag{2.22}$$

$$\vec{\nabla} \cdot \vec{u} = 0 \; . \tag{2.23}$$

This strong form of the problem will be solved numerically. Therefore, a certain computational methodology will have to be performed. The Stabilized Discontinuous Galerkin method is chosen, as it has proved its usefulness in the past [24]. In the next chapter, this computational methodology will be discussed in detail together with the numerical approach.

Chapter 3

Computational method

A considerable problem in the computation of viscoelastic flows, is the loss of convergence for high values of the Weissenberg number We. With increasing We, the elastic behaviour of the fluid becomes relatively more dominant related to the viscous effects. Therefore, different methods have been developed to obtain computational results with higher Weissenberg numbers.

In most of these methods, the stresses are taken as global unknowns. But since the flow of commercial polymer melts, which describe the rheological behaviour more realisticly with multiple relaxation modes, is in our interest, the number of degrees of freedom increases enormously in the numerical computations. Thus, a numerical method has been developed that eliminates the stresses at element level, only leaving the velocities as global degrees of freedom. This Discontinuous Galerkin method was originally developed by Lesaint and Raviart [6]. Fortin and Fortin [7] were the first to use the method for the analysis of viscoelastic flows.

Recently, Guénette and Fortin [27] proposed a stabilization modification for viscoelastic flows. They introduced an extra variable, which fits the discrete rate of deformation tensor in an extra equation. Their method leaves the constitutive equation untouched, in comparison to the EVSS method [4, 10], which separates the extra stress in viscous and elastic components.

Our method combines the Discontinuous Galerkin method by Fortin and Fortin [7] with the EVSS method by Rajagopalan *et al.* [4]. A stabilization, based on the method by Guénette and Fortin [27], is applied to this combination. We call the so obtained approach the Stabilized Discontinuous Galerkin method. In the next section, the method is discussed more in detail. Then, the problem is written in a weighted residual weak formulation to be able to apply the Spectral Element Method for the spatial discretization, explained in section 3.2. The next section contains the Gauss-type quadrature rules. These are used to compute the integrals numerically, originated from the weak formulation. The last section of this chapter covers the solution method.

3.1 Stabilized Discontinuous Galerkin method

In **Problem MP** (Eqn. (2.21) - (2.23)) an extra term is added to the momentum equation and an extra equation is added to fit a discrete rate of deformation tensor. Thus, a more

stable method is obtained. The stabilized strong formulation of the flow can now be written as:

Problem SMP *Find* $(\boldsymbol{\tau}_i, \vec{u}, p, \bar{\boldsymbol{D}})$ *such that*

$$We_i \mathcal{L}_i^* \boldsymbol{\tau}_i + \boldsymbol{Y}_i \cdot \boldsymbol{\tau}_i - 2\eta_i \boldsymbol{D}_u = \boldsymbol{0} \qquad i = 1, \ldots, M \ , \tag{3.1}$$

$$-\frac{1}{Re}\vec{\nabla} \cdot \left(-p\boldsymbol{I} + 2\eta_s \boldsymbol{D}_u + 2\beta \sum_{i=1}^{M} \eta_i (\boldsymbol{D}_u - \bar{\boldsymbol{D}}) + \sum_{i=1}^{M} \boldsymbol{\tau}_i\right) = \vec{0} \ , \tag{3.2}$$

$$\bar{\boldsymbol{D}} - \boldsymbol{D}_u = \boldsymbol{0} \ , \tag{3.3}$$

$$\vec{\nabla} \cdot \vec{u} = 0 \ , \tag{3.4}$$

where β is a dimensionless parameter and $\bar{\boldsymbol{D}}$ the discrete rate of deformation.

For a finite element approach, the strong formulation has to be converted to a weighted residual formulation. Therefore, the set of equations is projected on spaces still to be defined, by premultiplication with test functions forming the basis of the space and integration over the domain. This algorithm still contains second-order derivatives and the solution is limited to certain restrictions regarding the functional space. By now choosing test functions which are zero on the boundaries of the domain, and applying partial integration over the domain, a set remains with at most first-order derivatives and no boundary integrals. For it has only first-order derivatives, the numerical solution can be found in a functional space with less strict demands as compared to the strong form, which contains second-order derivatives. Thus, the obtained weighted residual formulation is called the weak formulation.

For the two dimensional case, the set of trial velocity, test velocity, test pressure, test stresses and test rate of deformation functions, denoted by resp. $\mathcal{U}$, $\mathcal{V}$, $\mathcal{Q}$, $\mathcal{S}$ and $\mathcal{G}$, are defined by:

$$\mathcal{U} = \{\vec{u} \mid \vec{u} \in [H^1(\Omega)]^2, \vec{u} = \vec{u}_\Gamma \ \text{ on } \ \Gamma\} \ , \tag{3.5}$$

$$\mathcal{V} = \{\vec{v} \mid \vec{v} \in [H^1(\Omega)]^2, \vec{v} = \vec{0} \ \text{ on } \ \Gamma\} \ , \tag{3.6}$$

$$\mathcal{Q} = \{q \mid q \in L^2(\Omega)\} \ , \tag{3.7}$$

$$\mathcal{S} = \{\boldsymbol{S} \mid \boldsymbol{S} \in [L^2(\Omega)]^{m \times m}\} \ , \tag{3.8}$$

$$\mathcal{G} = \{\boldsymbol{G} \mid \boldsymbol{G} \in [L^2(\Omega)]^{m \times m}\} \ , \tag{3.9}$$

where $L^2(\Omega)$ is the space of all square integrable functions over Ω and $H^1(\Omega)$ is the Sobolev space of functions with first-order derivatives which are square integrable over the domain:

$$L^2(\Omega) = \{u \in \Omega \mid \int_\Omega u^2 d\Omega < \infty\} \ , \tag{3.10}$$

$$H^1(\Omega) = \{u \in \Omega \mid u \in L^2(\Omega), \nabla u \in L^2(\Omega)\} \ . \tag{3.11}$$

In case of plane flow conditions: $m = 2$, in case of axisymmetric flow conditions: $m = 3$.
The weighted residual weak formulation then reads:

Problem SGW *Find* $(\boldsymbol{\tau}_i, \vec{u}, p, \bar{\boldsymbol{D}}) \in \mathcal{S}\times\mathcal{U}\times\mathcal{Q}\times\mathcal{G}$ *such that for all* $(\boldsymbol{S}_i, \vec{v}, q, \boldsymbol{G}) \in \mathcal{S}\times\mathcal{V}\times\mathcal{Q}\times\mathcal{G}$

$$(\boldsymbol{S}_i, We_i \mathcal{L}_i^* \boldsymbol{\tau}_i + \boldsymbol{Y}_i \cdot \boldsymbol{\tau}_i - 2\eta_i \boldsymbol{D}_u) = 0 \ , \tag{3.12}$$

$$-\Big(\frac{1}{Re}\boldsymbol{D}_v, 2\eta_s \boldsymbol{D}_u + 2\beta \sum_{i=1}^{M} \eta_i (\boldsymbol{D}_u - \bar{\boldsymbol{D}}) + \sum_{i=1}^{M} \boldsymbol{\tau}_i\Big) + (\frac{1}{Re}\vec{\nabla} \cdot \vec{v}, p) = 0 \ , \tag{3.13}$$

$$(\boldsymbol{G}, \bar{\boldsymbol{D}} - \boldsymbol{D}_u) = 0 \ , \tag{3.14}$$

$$(q, \vec{\nabla} \cdot \vec{u}) = 0 \ , \tag{3.15}$$

where $(\cdot,\cdot)$ denotes the L_2-inner product on the domain Ω and $2\boldsymbol{D}_v$ is again defined as $2\boldsymbol{D}_v = (\vec{\nabla}\vec{v}) + (\vec{\nabla}\vec{v})^c$.
The definition of the L_2-inner product is given as:

$$(\varphi, \psi) = \int_{\Omega} \varphi(\vec{x})\psi(\vec{x})w(\vec{x})d\Omega \ , \tag{3.16}$$

where $w(\vec{x})$ is the weight function for the natural inner product.

In order to tackle problems with complex geometries, the domain Ω is divided into K elements such that:

$$\Omega = \bigcup_{e=1}^{K} \Omega^e \quad \wedge \quad \bigcap_{i\neq j} \Omega^e = \emptyset \ . \tag{3.17}$$

To reduce the computation time, the extra stress $\boldsymbol{\tau}$ and the pressure p will be eliminated at element level. Therefore, a discontinuous discretization of the stress over the element interfaces is performed. For this, upwinding is performed by adding integrals on the element interfaces [28], using the Gauss theorem twice. So, a step in the stress is forced at the interfaces. The Gauss theorem is as follows:

$$\int_{\Omega^e} v\vec{u} \cdot \vec{\nabla}\varphi d\Omega = \int_{\Gamma^e} v\vec{u} \cdot \vec{n}\varphi d\Gamma - \int_{\Omega^e} \vec{\nabla} \cdot (\vec{u}v)\varphi d\Omega \ , \tag{3.18}$$

where Γ^e is the boundary of element Ω^e, $\vec{n}$ the unit outward normal to Γ^e, v a test function and φ the function to be discretized discontinuously.

Although the stresses are discretized discontinuously over the element interfaces, the stresses are still a global degree of freedom due to the integral over the inflow boundary of the element. By taking the stress of the neighbouring, upwinding element explicit and

the stresses belonging to the element itself implicit, a semi explicit/implicit method (SDGE) is accomplished. All variables are known except for the implicit stresses and therefore, the elimination of the stress variables at the element level can be performed. This provides an efficient handling of multiple modes (see Baaijens [29, 30]).
The dimensionless mixed weak formulation is now given by:

Problem SDGE *Find* $(\boldsymbol{\tau}_i, \vec{u}, p, \bar{\boldsymbol{D}}) \in \mathcal{S} \times \mathcal{U} \times \mathcal{Q} \times \mathcal{G}$ *such that for all* $(\boldsymbol{S}_i^d, \vec{v}, q, \boldsymbol{G}) \in \mathcal{S} \times \mathcal{V} \times \mathcal{Q} \times \mathcal{G}$

$$(\boldsymbol{S}_i^d, We_i\, \mathcal{L}_i^* \boldsymbol{\tau}_i + \boldsymbol{Y}_i \cdot \boldsymbol{\tau}_i - 2\eta_i \boldsymbol{D}_u)$$

$$-\sum_{e=1}^{K} \left[\int_{\Gamma_{in}^e} \boldsymbol{S}_i^d : We_i\, \vec{u} \cdot \vec{n} (\boldsymbol{\tau}_i - \boldsymbol{\tau}_i^{n^{ext}}) d\Gamma \right] = 0 \qquad i = 1, \ldots, M \ , \tag{3.19}$$

$$-\Big(\frac{1}{Re}\boldsymbol{D}_v, 2\eta_s \boldsymbol{D}_u + 2\beta \sum_{i=1}^{M} \eta_i (\boldsymbol{D}_u - \bar{\boldsymbol{D}}) + \sum_{i=1}^{M} \boldsymbol{\tau}_i\Big) + (\frac{1}{Re}\vec{\nabla} \cdot \vec{v}, p) = 0 \ , \tag{3.20}$$

$$(\boldsymbol{G}, \bar{\boldsymbol{D}} - \boldsymbol{D}_u) = 0 \ , \tag{3.21}$$

$$(q, \vec{\nabla} \cdot \vec{u}) = 0 \ , \tag{3.22}$$

where $(\cdot, \cdot)$ denotes the L_2-inner product on the domain Ω. $2\boldsymbol{D}_v$ is defined as $2\boldsymbol{D}_v = (\vec{\nabla}\vec{v}) + (\vec{\nabla}\vec{v})^c$ and $\boldsymbol{\tau}_i^{n^{ext}}$ is the explicit taken stress tensor at $t = t^n$ of the neighbouring, upwinding element. Although the part with the Deborah number De is neglected, it is still taken into account in the numerical program.

Now, we have a mixed weak formulation of the weighted residuals type with discontinuously discretized stresses. However, to solve these integrals numerically, discretization in a finite number of points has to be applied. This spatial discretization is dealt with in the next section.

3.2 Spectral Element Method

For complex problems, the integrals of **Problem SDGE** (Eqn. (3.19) - (3.22)) in general cannot be solved analytically. Therefore, numerical integration by Gauss-type quadrature rules in a finite number of integration points is applied. The different Gauss-type numerical quadrature rules are discussed in the next section. Here, the spatial discretization effectuated by a Spectral Element Method is treated. (See also Canuto *et al.* [17]). SEM is a specific application of the method of weighted residuals as used in our weak formulation. It combines the geometry flexibilities of the Finite Element Method with the high accuracy of the Spectral Method (Patera [31]).
In general form, the weighted residual formulation is given by:

$$(\pounds u - f, v) = 0 \qquad \forall\, v \in \mathcal{V} \ , \tag{3.23}$$

where $\mathcal{L}$ is a elliptic differential operator. The solution of the problem u can be written as a series expansion of the basis functions φ_i:

$$u(x) = \sum_{i=0}^{\infty} c_i \varphi_i \ , \qquad u \in \mathcal{U} \ , \tag{3.24}$$

where c_i are the unknown expansion coefficients. The Galerkin method is obtained by choosing the same basis for the discrete test functions v:

$$v(x) = \sum_{j=0}^{\infty} \varphi_j \ , \qquad v \in \mathcal{V} \ . \tag{3.25}$$

An exponential (or spectral) decay of the expansion coefficients c_i can be obtained for test functions φ_i that are eigenfunctions of singular Sturm-Louiville problems defined on $\Omega = (-1, 1)$ (Canuto *et al.* [17]). In general, polynomial solutions of singular Sturm-Louiville problems are Jacobi polynomials, like Chebyshev and Legendre polynomials (see Canuto *et al.* [17]). The Chebyshev and Legendre polynomials are orthonormal with respect to a weight function w:

$$\int_{-1}^{1} \varphi_i(x)\varphi_j(x)w(x)dx = \delta_{ij} \ , \tag{3.26}$$

with δ_{ij} denoting the Kronecker symbol. The weights for the Chebyshev and Legendre polynomials are $w(x) = (1 - x^2)^{-1/2}$ and $w(x) = 1$, respectively.
In practice, an approximate solution $P_N u$ is used in a finite dimensional subspace $\mathcal{U}^N \subset \mathcal{U}$, represented by the truncated series expansion of the basis functions φ_i:

$$P_N u(x) = \sum_{i=0}^{N} c_i \varphi_i \ . \tag{3.27}$$

For the weighting functions, the same truncated series expansion is taken, leaving the Galerkin discrete weighted residual formulation as:

$$(\mathcal{L} P_N u - f, v) = 0 \qquad \forall\, v \in \mathcal{V}^N \ . \tag{3.28}$$

Rewriting Eqn. (3.28) gives:

$$\int_{\Omega} \sum_{i=0}^{N} c_i \mathcal{L}\varphi_i \varphi_j w d\Omega = \int_{\Omega} f \varphi_j w d\Omega \ , \qquad j = 0, \ldots, N \ , \qquad (\mathcal{V}^N = \{\varphi_j\}_{j=0}^{N}) \ . \tag{3.29}$$

This is called the Spectral Galerkin method and in a matrix formulation this is given by:

$$\underset{\sim}{S}\, \underset{\sim}{c} = \underset{\sim}{f} \ , \tag{3.30}$$

with

$$S_{ij} = \int_{\Omega} \mathcal{L}\varphi_j \varphi_i w d\Omega \tag{3.31}$$

and

$$f_j = \int_{\Omega} f\varphi_j w d\Omega \ . \tag{3.32}$$

Note, that $\underline{c}$ is the vector containing the unknowns c_i. The approximate solution $P_N u$ can be computed using Eqn. (3.27).
Because the differential operator $\pounds$ may contain terms, which cannot be integrated analytically, numerical integration has to be applied. As to conserve the accuracy of the higher order polynomials, higher order Gauss-type numerical quadrature rules in a finite number of integration points is performed. Also, Legendre polynomials are chosen, for they have a weight function $w(x) = 1$. Because w is independent of x, partial integration is easier to perform and the numerical integration is more exact.
If x_k and w_k are the integration points and weights of the Gauss-type numerical quadrature rule, the elements of the matrix and right hand side of the Spectral Galerkin method read:

$$S_{ij} = \sum_{k=0}^{N} \pounds\varphi_j(x_k)\varphi_i(x_k)w_k \tag{3.33}$$

and

$$f_j = \sum_{k=0}^{N} f(x_k)\varphi_j(x_k)w_k \ . \tag{3.34}$$

The basis with basis functions φ_i can be transformed to a basis with basis functions ϕ_i that defines the same finite dimensional subspace $\mathcal{U}^N$ and also holds the property:

$$\phi_i(x_k) = \delta_{ik} \ . \tag{3.35}$$

The elements (3.33) and (3.34) can now be computed efficiently and thus reducing the computational time and data storage.

As given before, the domain Ω is divided into K non-overlapping subdomains like in the Finite Element Method. Per element with $\xi \in [-1, 1]$, the Lagrange interpolants ϕ_i through the local Legendre-Gauss-Lobatto points are defined by the following relationship:

$$\phi_i(\xi) = -\frac{(1-\xi^2)L'_N(\xi)}{N(N+1)L_N(\xi_k)(\xi - \xi_k)} \ , \tag{3.36}$$

where ξ_k are the Legendre-Gauss-Lobatto collocation points and $L_N(\xi)$ is the Legendre polynomial of degree N:

$$L_0(\xi) = 1 \ , \qquad L_N(\xi) = \frac{1}{2N \cdot N!}\frac{d^N[(\xi^2-1)^N]}{d\xi^N} \ . \tag{3.37}$$

The approximate solution $\Pi_N^e u$ per element can now be written as a linear combination of Lagrange interpolation polynomials through the local Gauss-type quadrature points:

$$\Pi_N^e u(x) = \sum_{i=0}^{N} u_i \phi_i(\xi) \ . \tag{3.38}$$

In this way, the coefficients are just given by the value of the function in the interpolation point $u_i = \Pi_N^e u(x_k)$.
Still, an exponential convergence in the interpolation error for infinitely smooth functions is obtained, by increasing the degree of the polynomial N (Canuto *et al.* [17]). Algebraic convergence can be obtained by augmenting the number of elements.

The extension of the Spectral Element Method towards two- and three-dimensional problems is straightforward. More dimensional basis functions are constructed as a tensor product of the one-dimensional basis functions. For the two-dimensional case, the series expansion for the approximate solution can be written as:

$$\Pi_N^e u(x,y) = \sum_{i=0}^{N}\sum_{j=0}^{N} u_{ij}\phi_i(\xi)\phi_j(\eta) \ , \tag{3.39}$$

where ξ and η are the local coordinates on the interval $[-1,1]$, corresponding to the global coordinates x and y on the domain Ω^e.
For the Legendre-Gauss-Lobatto points, the interpolants still satisfy the following property:

$$\phi_i(\xi_k) = \delta_{ik} \qquad \forall\, i,k \in [0,N]^2 \ . \tag{3.40}$$

The spatial discretization is now accomplished. The next step involves the numerical integration and the choice of the associated quadrature rules.

3.3 Numerical integration

In order to obtain results with satisfying accuracy for numerical integration using high order polynomials, also high order quadrature rules have to be applied. The two most important classes are the Legendre-Gauss (LG) and the Legendre-Gauss-Lobatto (LGL) quadrature rules. LG integration is exact up to polynomials of the order $2N$. The functions are then evaluated in the Legendre-Gauss points. LGL integration, based on integration points that coincide with the interpolation points, is only exact up to polynomials of the order $2N-2$. For low order methods, this last quadrature rule may lead to inadmissible loss of accuracy. However, for high order methods, the loss of accuracy is relatively low ($2N-2 \mapsto 2N$ for $N \mapsto \infty$). With the LGL quadrature rule, property (3.40) can be used optimally and especially in more dimensions, the gain in computing time and data storage is enormous.

Now, numerical integration has to be applied on the set of L_2-inner products (3.19) to (3.22). However, as the basis functions are defined on the interval $[-1,1]^2$, first, transforming the global coordinates (x,y) to the local coordinates (ξ,η) is necessary. Per element, the transformation of a L_2-inner product can be written as:

$$(\varphi,\psi)^e = \int_{\Omega^e} \varphi(x,y)\psi(x,y)w(x,y)d\Omega^e = \int_{-1}^{1}\int_{-1}^{1} \varphi(\xi,\eta)\psi(\xi,\eta)w(\xi,\eta)J\Big(x(\xi,\eta),y(\xi,\eta)\Big)d\xi d\eta \ . \tag{3.41}$$

The global coordinates (x, y) correspond to the element domain Ω^e. The local coordinates (ξ, η) are defined on the interval $[-1, 1]^2$ and, depending on the type of quadrature rule, are the LG or LGL points. The Jacobian J accomplishes the transformation from the global domain Ω^e to the local interval $[-1, 1]^2$, see figure 3.1, and is defined as:

$$J\Big(x(\xi,\eta), y(\xi,\eta)\Big) = \frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta}\frac{\partial y}{\partial \xi} \ . \tag{3.42}$$

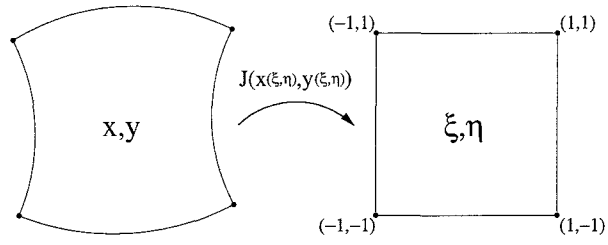

Figure 3.1: *Transformation from global to local coordinates.*

After application of the numerical integration, a L_2-inner product reads:

$$(\varphi, \psi)^e \approx \sum_{i=0}^{N}\sum_{j=0}^{N} \varphi(\xi_i, \eta_j)\psi(\xi_i, \eta_j)w(\xi_i, \eta_j)J(\xi_i, \psi_j) \ . \tag{3.43}$$

Going back to **Problem SDGE** (Eqn. (3.19) - (3.22)), notice, as in the finite element context, that the velocity and pressure discretizations can not be chosen independently. They must verify the Babuška-Brezzi condition in order to avoid the presence of spurious pressure modes. This theory was set up by Maday and Patera [32] in the framework of the spectral element method. They showed that an optimal choice, related to the associated quadrature rules, implies velocity nodes on a Legendre-Gauss-Lobatto (LGL) grid of the order N in both spatial directions and that the pressure is attached to a Legendre-Gauss (LG) grid of the order $N - 2$, also in both spatial directions. This means that the pressure has no discrete points on the spectral LGL grid and all the points are on the inside of the element. Thus, the pressure is discontinuous over the elements.
The discretization for the extra stress $\boldsymbol{\tau}_i$ and the discrete rate of deformation $\bar{\boldsymbol{D}}$ is still free to choose. The extra stress is chosen on the LGL grid of the order $N - 1$ in both spatial directions as to ensure the compatibility between the elastic and viscous part of the stress (Talwar and Khomami [16]). A disadvantage is that the stress need to be interpolated to the velocity LGL grid for integration of the momentum and constitutive equations. Thus, the computation time is increased as property (3.40) is not met. However, as the extra stress is solved at element level, this disadvantage is not of a high priority.
The rate of deformation $\boldsymbol{D}$ is discontinuous over the element boundaries and of an order one less than the velocity. In order to obtain a more fluid approach of the rate of deformation, the discrete rate of deformation $\bar{\boldsymbol{D}}$ is chosen to be continuous over the element boundaries. As only one grid is present, the discretization is of the same order as the velocity.
Denote with $Q_{N,K}$ the space of all polynomials of degree N or less, defined over the K quadrilateral elements. Further, $(\cdot)^d$ defines a discontinuous discretization over the element interfaces. In short, the discretization can now be written as:
$(\boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p) \mapsto (Q^d_{N-1,K}, Q_{N,K}, Q_{N,K}, Q^d_{N-2,K})$.

For integration, the quadrature rule defined on the LG grid of the order N is used. This means, that interpolation of the quantities on the LGL grid onto the LG grid is necessary. The pressure term and the test functions associated with the pressure are defined on the LG grid of the order $N-2$ and therefore need to be extrapolated to the LG grid of the order N. Firstly, the LG pressure points are extrapolated to the LGL grid of the order $N-2$ using the inverse values of the interpolants of the order $N-2$ in these LG pressure points. Secondly, interpolation of these quantities onto the LG grid of the order N is performed.
As can easily be seen, the advantage of property (3.40) is lost using the LG quadrature rule and thus the computation time is significantly increased. However, a program, based on the Finite Element Method, was adjusted to accomplish a faster implementation of the Spectral Element Method for the computation of viscoelastic flows. In that program, an efficient use of property (3.40) was not easily met and thus loosing the advantage.

In future, for high order integration, the computation time is easily decreased after efficient programming by applying the LGL quadrature rule. Still, the pressure term and the test functions associated with the pressure, need to be extrapolated to the LGL grid.

For an axisymmetric case, Van Kemenade and Deville [15] pointed out some adjustments. As in the axisymmetric case the equations are expressed in cylindrical coordinates (r, z, θ), geometric factors $\frac{1}{r}$ are introduced. Most of these factors disappear by the definition of the infinitesimal volume $d\Omega = 2\pi r dr dz$. Thus, the factor $\frac{1}{r}$ is only present in the momentum equation. However, as this equation satisfies the essential condition $u_r = 0$ on the symmetry axis, the term containing the factor is not computed at $r = 0$.
As far as the constitutive relationships are concerned, they generate trivial relations $(0 = 0)$ on the symmetry axis. This results in an undetermined system of equations. Some additional relationships, following from the symmetry conditions, associated with the extra-stress unknowns on the symmetry axis are needed:

$$\tau_{rz} = 0\,, \qquad \tau_{\phi\phi} = \tau_{rr}\ . \tag{3.44}$$

The two remaining equations for τ_{rr} and τ_{zz} are built by dividing the original relations by r.

So far, the Stabilized Discontinuous Galerkin method is explained, the discretizations for the different variables is accomplished and choices for the numerical integrations are made. Applying the Gauss-type numerical quadrature rules on Eqn. (3.19) - (3.22), leaves us a set of discrete equations to be solved. Thus, the next logic step is to distinguish the solution strategy.

3.4 Solution method

The set of equations (3.19) to (3.22) is obviously non-linear. It is necessary to carry out a linearization process in order to obtain a set of linear algebraic equations. Here, a Newton's linearization will be carried out.
Denote $\delta\boldsymbol{\tau}_i$, $\delta\vec{u}$, $\delta\bar{\boldsymbol{D}}$ and δp as the variations of the extra stress tensor of a mode, the velocity field, the discrete rate of deformation tensor and the pressure, respectively. And define the short notation for Eqn. (3.19) as $F_i(\boldsymbol{S}_i^d, \vec{v}, \boldsymbol{G}, q; \boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p)$, for Eqn. (3.20) as $G(\boldsymbol{S}_i^d, \vec{v}, \boldsymbol{G}, q; \boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p)$, for Eqn. (3.21) as $E(\boldsymbol{S}_i^d, \vec{v}, \boldsymbol{G}, q; \boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p)$ and for Eqn. (3.22) as

$H(\boldsymbol{S}_i^d, \vec{v}, \boldsymbol{G}, q; \boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p)$. Assuming that the directional derivatives of the above mentioned equations exist, the definition for a derivative is given as:

$$\delta_u F_i(\boldsymbol{S}_i^d, \vec{v}, \boldsymbol{G}, q; \boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p; \delta\vec{u}) = \lim_{\theta \to 0} \frac{F_i(\boldsymbol{S}_i^d, \vec{v}, \boldsymbol{G}, q; \boldsymbol{\tau}_i, \vec{u} + \theta\delta\vec{u}, \bar{\boldsymbol{D}}, p) - F_i(\boldsymbol{S}_i^d, \vec{v}, \boldsymbol{G}, q; \boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p)}{\theta} . \quad (3.45)$$

Now, in a matrix formulation, the linearized form of the set of equations (3.19) to (3.22) is given by:

$$\begin{bmatrix} \delta_\tau \underline{F}_i & \delta_u \underline{F}_i & \delta_{\bar{D}} \underline{F}_i & \delta_p \underline{F}_i \\ \delta_\tau \underline{G} & \delta_u \underline{G} & \delta_{\bar{D}} \underline{G} & \delta_p \underline{G} \\ \delta_\tau \underline{E} & \delta_u \underline{E} & \delta_{\bar{D}} \underline{E} & \delta_p \underline{E} \\ \delta_\tau \underline{H} & \delta_u \underline{H} & \delta_{\bar{D}} \underline{H} & \delta_p \underline{H} \end{bmatrix} \begin{bmatrix} \delta\underset{\sim}{\tau}_i \\ \delta\underset{\sim}{u} \\ \delta\underset{\sim}{\bar{D}} \\ \delta\underset{\sim}{p} \end{bmatrix} = \begin{bmatrix} -\underset{\sim}{F}_i \\ -\underset{\sim}{G} \\ -\underset{\sim}{E} \\ -\underset{\sim}{H} \end{bmatrix} . \quad (3.46)$$

The different parts of set (3.46) are written out in Appendix A.
After elimination of the extra stress and the pressure, the resulting set of linear equations is:

$$\begin{bmatrix} \delta_u \underline{G} - \delta_\tau \underline{G} \cdot (\delta_\tau \underline{F}_i)^{-1} \cdot \delta_u \underline{F}_i - \delta_p \underline{G} \cdot (\delta_p \underline{H})^{-1} \cdot \delta_u \underline{H} & \delta_{\bar{D}} \underline{G} \\ \delta_u \underline{E} & \delta_{\bar{D}} \underline{E} \end{bmatrix} \begin{bmatrix} \delta\underset{\sim}{u} \\ \delta\underset{\sim}{\bar{D}} \end{bmatrix} = \begin{bmatrix} -\underset{\sim}{G} + \delta_\tau \underline{G} \cdot (\delta_\tau \underline{F}_i)^{-1} \cdot \underset{\sim}{F}_i + \delta_p \underline{G} \cdot (\delta_p \underline{H})^{-1} \cdot \underset{\sim}{H} \\ -\underset{\sim}{E} \end{bmatrix} . \quad (3.47)$$

Because one of the matrices to be inverted is zero ($\delta_p \underline{H} = 0$), a penalty parameter in the derivative of the mass matrix to the pressure is introduced:

$$\delta_p \underline{H} = \int_\Omega q\kappa\delta p d\Omega . \quad (3.48)$$

Therefore, inverting this matrix is no longer a problem. This penalty method adds compressibility to the problem and thus a different solution is obtained than actually intended.

The linear set (3.47) is solved by Gaussian elimination with a direct solver using the LU-decomposition without partial pivoting. Firstly, the large matrix $\underline{A}$ of the problem $\underline{A}\,\underset{\sim}{x} = \underset{\sim}{b}$ is decomposed in $\underline{A} = \underline{L}\,\underline{U}$, where $\underset{\sim}{x}$ is the approximate solution, $\underset{\sim}{b}$ the right hand side vector and $\underline{L}$ a lower triangular and $\underline{U}$ a unit upper triangular matrix. Secondly, the approximate solution $\underset{\sim}{x}$ is found by forward and backward substitution in $\underline{L}\,\underset{\sim}{y} = \underset{\sim}{b}$ and $\underline{U}\,\underset{\sim}{x} = \underset{\sim}{y}$ respectively. Then, the velocity and discrete rate of deformation variables are updated according to:

$$\underset{\sim}{u}^{n+1} = \underset{\sim}{u}^n + \delta\underset{\sim}{u}; \qquad \underset{\sim}{\bar{D}}^{n+1} = \underset{\sim}{\bar{D}}^n + \delta\underset{\sim}{\bar{D}} . \quad (3.49)$$

At element level, the extra stress and pressure unknowns are solved using the NAG Fortran Library subroutine F04ARF. This subroutine determines the approximate solution by Crout's factorization with partial pivoting: $\underline{P}\,\underline{A} = \underline{L}\,\underline{U}$, where $\underline{P}$ is a permutation matrix. The solution is found with the next systems:

$$\delta_\tau \underline{F}_i \;\; \delta\underset{\sim}{\tau}_i = -\underset{\sim}{F}_i - \delta_u \underline{F}_i \;\; \delta\underset{\sim}{u} , \quad (3.50)$$

$$\delta_p \underline{H} \;\; \delta\underset{\sim}{p} = -\underset{\sim}{H} - \delta_u \underline{H} \;\; \delta\underset{\sim}{u} . \quad (3.51)$$

Afterwards, the variables are updated according to:

$$\underset{\sim}{\tau}_i^{n+1} = \underset{\sim}{\tau}_i^n + \delta\underset{\sim}{\tau}_i; \qquad \underset{\sim}{p}^{n+1} = \underset{\sim}{p}^n + \delta\underset{\sim}{p} \; . \tag{3.52}$$

These updated variables are used to determine the new large matrix $\underline{A}$ and right hand side $\underset{\sim}{b}$ of the linear set (3.47). Newton iteration is performed until the difference between the old solution and the new solution of all the variables are converged to a value smaller than a specific user defined convergence parameter.

As the computational method is explained and all the choices are made, the next chapter will contain some flow problems.

Chapter 4

Results and discussion

In order to investigate the performance of the Stabilized Discontinuous Galerkin numerical method, some test flows are calculated. First, a simple Poisseuille flow under Stokes' conditions is computed to check the rate of convergence of the method. Then, the benchmark problem of a falling sphere in a tube is considered and compared to the results of Lunsmann *et al.* [20] and Selen [24].

4.1 Stokes flow

In this section, a two-dimensional steady Stokes flow through a plane channel and an axisymmetric tube is considered. This problem is studied to investigate the spatial convergence of the method. The Stokes problem is recovered by selecting $\lambda_i = 0$ and $\eta_i = 0 \ \forall \ i = 1, \ldots, M$ in the Eqn. (3.19) - (3.22).
The convergence of the spectral element method can be achieved by increasing the polynomial degree, resulting in an exponential convergence when the problem is smooth enough [17, 32], or by augmenting the number of elements which results in an algebraic rate of convergence, as for the classical finite element technique.

As a test problem, a simple flow is used, also applied by Rønquist in his PhD thesis [33]. The equations for the analytical solution give:

$$\vec{u} = (u_1, u_2) = (1 - x_2^2, 0) \ , \tag{4.1}$$

$$p = \sin \pi x_1 \cdot \sin \pi x_2 \ , \tag{4.2}$$

$$\vec{f} = (f_1, f_2) = (2 + \pi \cos \pi x_1 \cdot \sin \pi x_2, \pi \sin \pi x_1 \cdot \cos \pi x_2) \ , \tag{4.3}$$

on the domain $\Omega = [-1, 1]^2$ with $\vec{u}$ specified on the domain boundary. A vector plot of the velocity and a streamline plot of the pressure in the 2D Cartesian plane strain case is shown in figure 4.1. These plots are generated by using the discretization $(\boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p) \ \mapsto$ $(Q^d_{1,16}, Q_{7,16}, Q_{1,16}, Q^d_{5,16})$.

As a first approach to achieve convergence, the order of the elements used to solve this problem is kept constant and the number of elements is increased. The discretization for

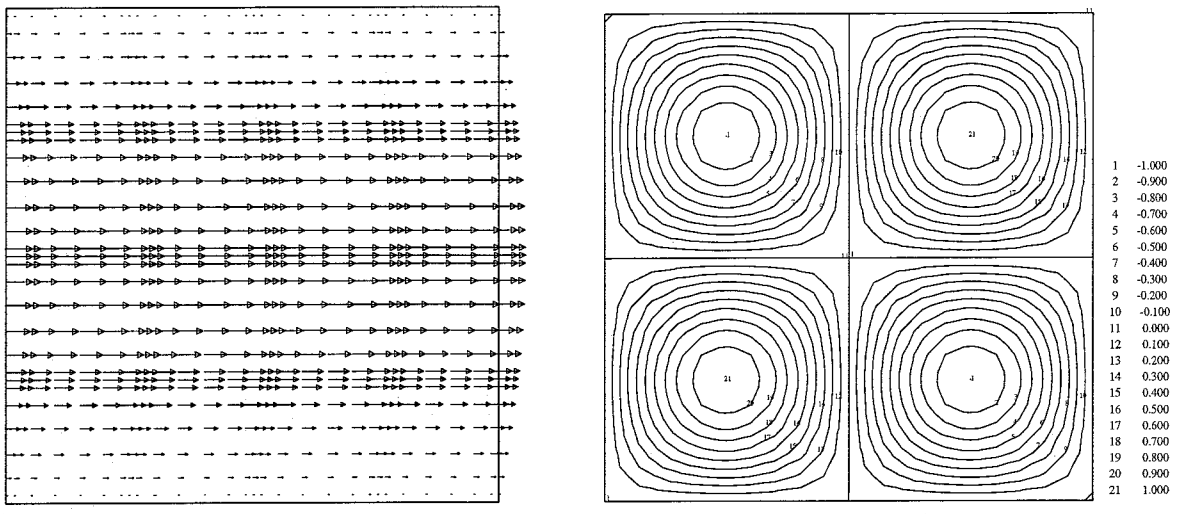

Figure 4.1: *Simple steady Stokes flow through a plane channel; left: velocity; right: pressure.*

each element is taken $(\boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p) \mapsto (Q^d_{1,K}, Q_{2,K}, Q_{1,K}, Q^d_{1,K})$. The interpolation of the extra stress and velocity gradient is taken bilinear, to keep the CPU-time as low as possible. In figure 4.2, the norm of the error between the analytical and the numerical solution as a function of the total number of degrees of freedom of the plane strain and axisymmetric case, respectively, is shown. The used norm is as follows:

$$\| u_{exact} - u_h \| = \sqrt{\frac{\sum_{i=1}^{N} (u_{exact_i} - u_{h_i})^2}{N}} \,. \tag{4.4}$$

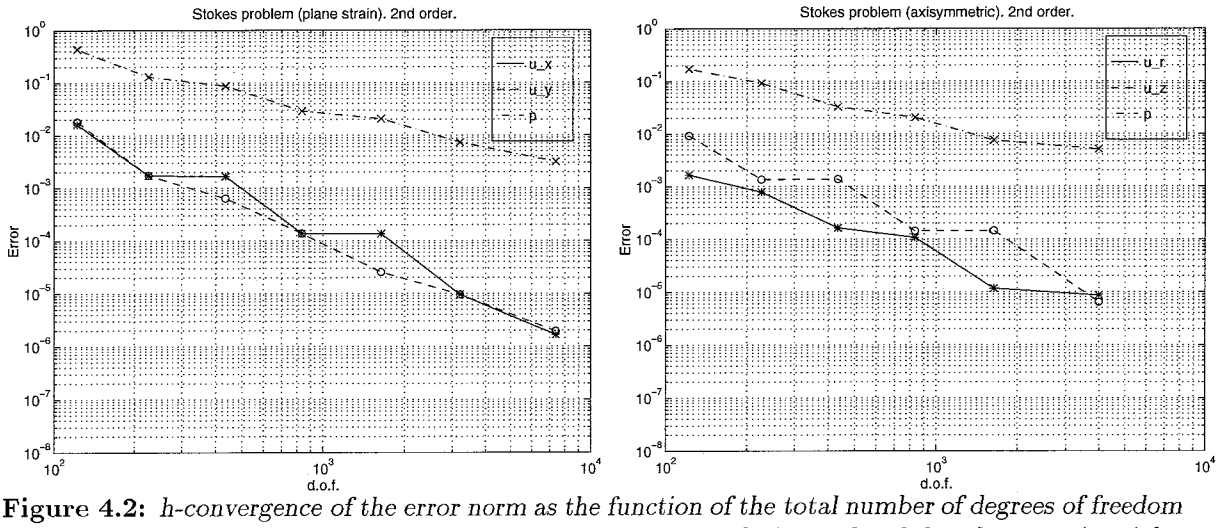

Figure 4.2: *h-convergence of the error norm as the function of the total number of degrees of freedom of the velocity and pressure for a certain interpolation order; left: plane strain; right: axisymmetric.*

As can easily be seen and was expected, convergence is achieved by increasing degrees of freedom in both the plane strain as the axisymmetric case. Note that the error in the pressure is larger than in the velocities. The fact that the pressure is one order lower than the velocities causes this difference. Probably the penalty parameter and the extrapolation of the pressure in the LG points onto the LGL grid has its influence as well. This last is necessary, because

for the graphical presentation and the determination of the norm of the error, all values need to be known in the Nth order LGL grid. Only one mesh is present, defined in this LGL grid and therefore, the pressure needs to be projected on the LGL grid, using Eqn. (3.39).

For a second approach, the domain is broken up into 32 spectral elements for the plane strain case and 16 elements for the axisymmetric case and the value of the polynomial degree is increased. The velocity ranges from second to seventh order interpolation and the pressure from bilinear to fifth order. Again, the norm as a function of the total number of degrees of freedom for both the plane strain as well as the axisymmetric case is plotted, see figure 4.3. Here, an exponential convergence is detected. Note also, that in the axisymmetric case, a difference can be detected between the errors for the velocity in the r- and the z-direction. The extra boundary conditions on the symmetry axis may explain this difference.

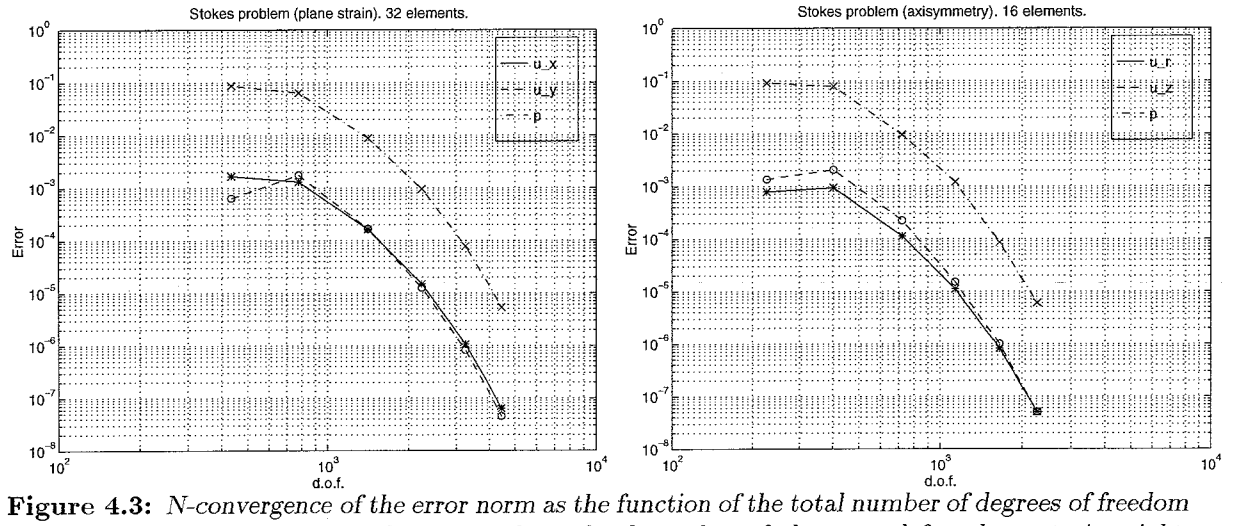

Figure 4.3: *N-convergence of the error norm as the function of the total number of degrees of freedom of the velocity and pressure for a fixed number of elements; left: plane strain; right: axisymmetric.*

To see the improvement of higher order elements on the solution, the error norm as a function of the polynomial degree N for a specific number of degrees of freedom is investigated. For every degree, approximately 2500 degrees of freedom are taken. The results of the plane strain case are given in figure 4.4 and table 4.1.

order	# nodes	# elm	# dof	Error u_1	Error u_2	Error p
2nd	825	192	2418	1.0374E-5	2.8751E-5	1.0420E-2
3rd	1000	104	2416	2.4100E-4	2.4406E-4	1.6614E-2
4th	957	56	2418	2.4043E-5	2.2406E-5	1.7660E-3
5th	966	36	2508	1.2679E-5	1.4953E-5	9.4690E-4
6th	925	24	2450	8.2234E-7	1.0520E-6	7.9758E-5
7th	841	16	2258	7.5491E-8	7.5491E-8	8.3208E-6

Table 4.1: *Numerical results of the simple steady Stokes flow problem.*

Again, after a drawback from the second to the third order, convergence can be noticed. But, only at a higher than fifth order interpolation, improvement of the velocity approximation is achieved. An improvement of the pressure approximation is already achieved at the fourth order interpolation.

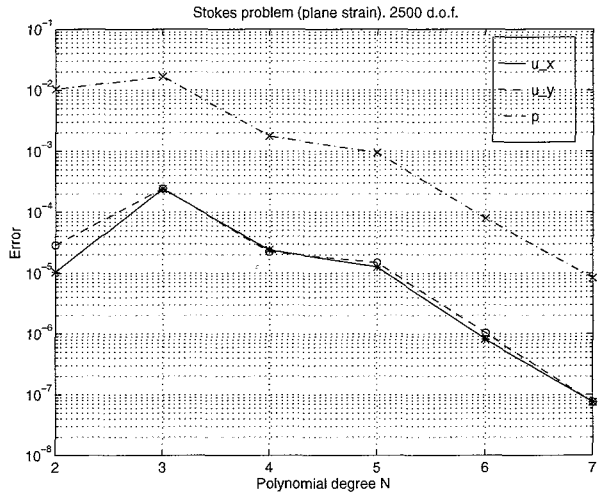

Figure 4.4: *Error norm as the function of the polynomial degree N for a fixed number of degrees of freedom.*

The drawback may be caused by the strange transition for the order of the polynomials in the velocity and the pressure. Where as the order of the velocity is increased by one from the second to the third order, the order of the pressure stays the same. A secondary reason could be that the velocity is independent of the x_1-direction (see Eqn. (4.1)) and therefore a refinement of the spatial discretization in that direction does not necessarily improve the solution of the velocity.

As can be detected from the results of the Stokes problem, this Stabilized Discontinuous Galerkin Method in combination with the Spectral Element Method serves its purpose. For simple problems, increasing the polynomial order improves the accuracy of the solution in an exponential convergence rate. In the next section, the performance of the method will be investigated on a more complex problem, the falling sphere in a tube.

4.2 Falling sphere in a tube

The benchmark problem of steady flow past a falling sphere in a tube is examined to test the performance of the numerical method. The Upper Convected Maxwell (UCM) constitutive model is applied, for this is believed to be the most difficult model to handle. As well, this model is often used in the literature. So, a comparison with results from the literature can be made. In all the calculations in this section only a single mode is used. The characteristic numbers are all scaled to unity, except for the relaxation time λ. And no source term will be present: $\vec{f} = \vec{0}$.
A flow of a viscoelastic fluid around a sphere of radius R_s falling with a uniform velocity U along the symmetry axis of a tube of radius R_t is considered. In the numerical simulations it is assumed that the sphere is motionless and the wall moves with velocity U in the positive z-direction instead. On the entry and exit sections the same uniform axial velocity U is imposed and the stresses are zero on the entry section. The flow is axisymmetric and therefore cylindrical coordinates (r, z, θ) are employed. The upstream section has a length of $12R_s$, the downstream section of $18R_s$. The flow domain is given in figure 4.5.

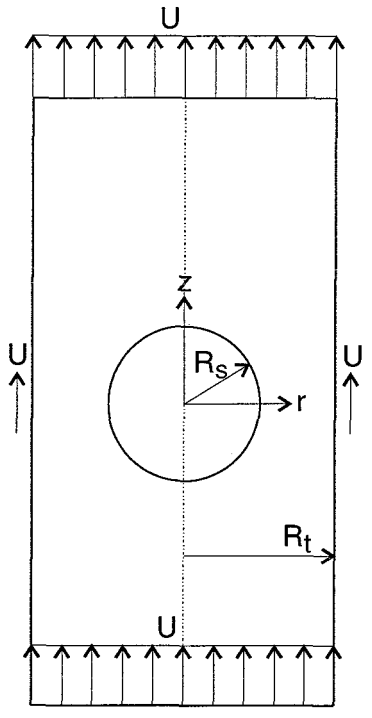

Figure 4.5: *Falling sphere in a tube benchmark problem.*

We define the ratio of the tube R_t and the sphere radius R_s as:

$$\chi = \frac{R_t}{R_s} \, . \tag{4.5}$$

In this report, only results with $\chi = 2$ will be presented, although the variation of the drag on the sphere with χ is also of interest. The Weissenberg number We, denoting the importance of the elasticity in the flow, is for this benchmark defined as:

$$We = \frac{\lambda U}{R_s} \, . \tag{4.6}$$

A quantity of interest for this benchmark problem, is the drag F exerted by the fluid on the sphere. For an unbounded Newtonian medium the drag is given by:

$$F = 6\pi\eta_s R_s U \, . \tag{4.7}$$

Lunsmann *et al.* [20] introduced a wall correction factor $K(\chi)$ given by:

$$K(\chi) = \frac{F(\chi)}{6\pi\eta_s R_s U} \, . \tag{4.8}$$

An approximation of this wall correction factor for a sphere falling through a tube filled with a Newtonian fluid was derived by Happel and Brenner [34], and is given by:

$$\eta_s R_s U K(\chi) \cong (1 - 2.10444\chi^{-1} + 2.08877\chi^{-3} - 0.94813\chi^{-5} - 1.372\chi^{-6} + 3.87\chi^{-8} - 4.19\chi^{-10} + \ldots)^{-1} \, . \tag{4.9}$$

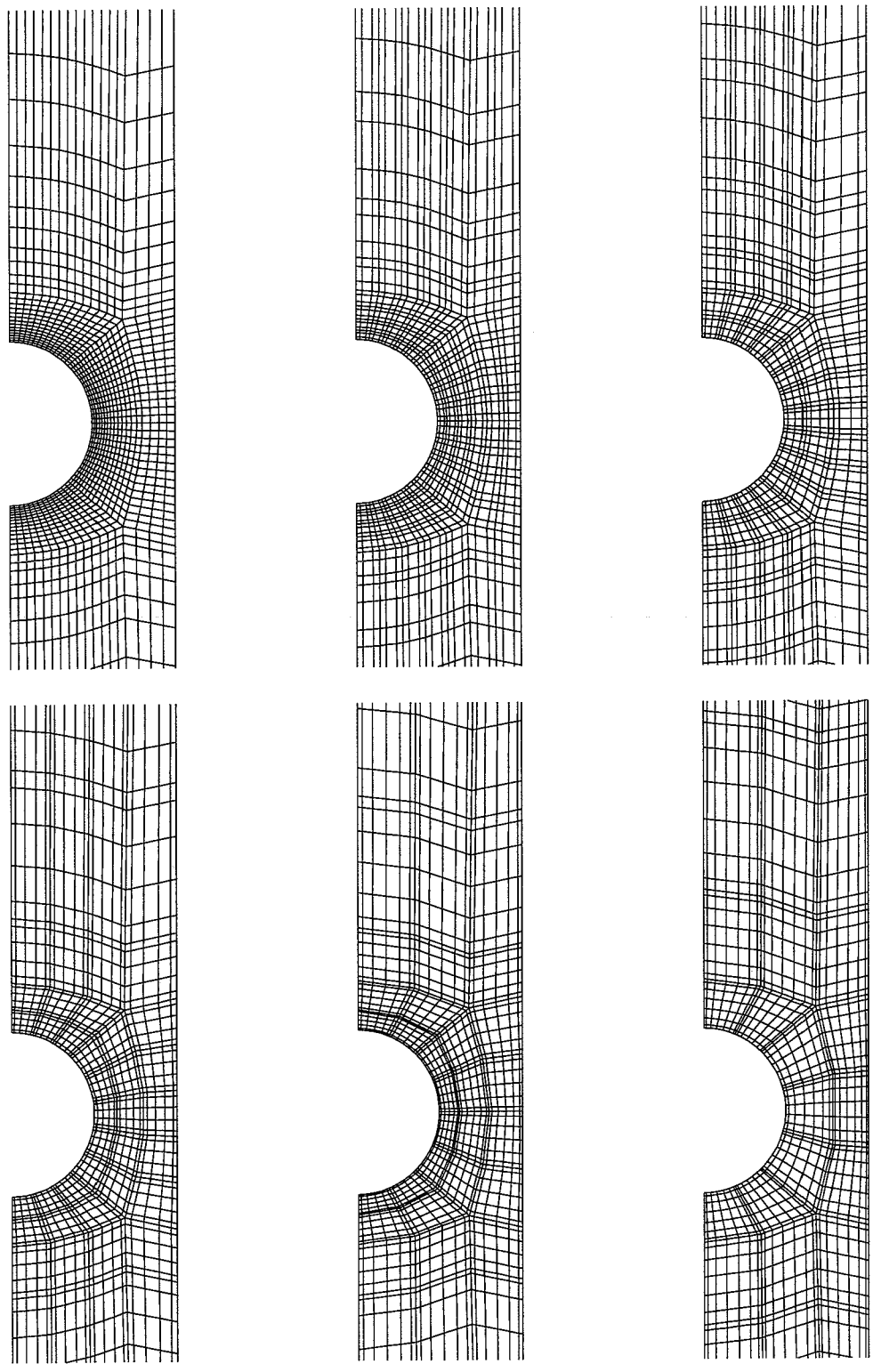

Figure 4.6: *Finite element representation of the spectral meshes used for the analysis of the falling sphere test; top: from left to right: M1, M2 and M3; bottom: from left to right: M4, M5 and M6.*

This expression is only valid for $\chi^{-1} \ll 1$, but gives quite accurate results for $\chi = 2$ according to Lunsmann *et al.* [20]. For $\chi = 2$, the analytical wall correction factor with equation (4.9) is $K = 5.922940$. Numerically, the drag is determined by integrating the total stress over the surface of the sphere.

Meshes are generated with approximately the same number of degrees of freedom for the different orders of the polynomials, again, in order to detect the improvement of higher order elements. In the sequel, to indicate the spectral discretization of the elements, the order of the velocity polynomials N is given. So, $N = 4$ means a discretization of $(\boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p) \mapsto (Q^d_{3,K}, Q_{4,K}, Q_{4,K}, Q^d_{2,K})$.
For every order, one mesh with approximately 8000 degrees of freedom is generated and the highest spectral discretization is $N = 7$. Figure 4.6 depicts the section of the finite element meshes near the sphere for the different spectral discretizations. Mesh M1, M2, M3, M4, M5 and M6 refer respectively to $N = 2$, $N = 3$, $N = 4$, $N = 5$, $N = 6$ and $N = 7$. The data concerning their characteristics are given in table 4.2.

mesh	order	# elements	# nodes	# dof	K (Newtons)
M1	2nd	368	1579	7895	5.946067
M2	3rd	165	1591	7955	5.928932
M3	4th	92	1589	7945	5.950521
M4	5th	59	1586	7930	5.935534
M5	6th	41	1585	7925	5.911481
M6	7th	31	1632	8160	5.903241

Table 4.2: *Characteristics of the meshes used for the analysis of falling sphere problem.*

First, the numerical results at Newtonian flow conditions will be investigated for relative accuracy of the method. Subsequently, the viscoelastic flows will be examined.

4.2.1 Newtonian flow

For $We = 0.0$ and $N = 2$, the contourplots of both the velocity and stress components as well as the pressure are shown in figure 4.7. Symmetry in the plane $z = 0$ occurs in all the components. Also, the maximum and minimum values of the u_r, p, τ_{rr}, τ_{zz} and $\tau_{\theta\theta}$ variables are symmetrical in this plane. From the u_z plot, the acceleration of the fluid around the sphere can be detected.

In figure 4.8 the wall correction factor K is plot as a function of the polynomial degree N. The straight dash-dotted line is the analytical determined value of K with equation (4.9) and the dashed line represents the numerically obtained value of K by Lunsmann *et al.* [20] and Selen [24]: $K = 5.9474$.
No convergence to a specific value can be detected for the Newtonian wall correction factor. For higher degrees, the mesh near the surface of the sphere is not fine enough to capture the flow sufficiently. For mesh M1, the Newtonian wall correction factor K is near to the values found in the literature. Most probably, if this mesh M1 is taken and the polynomial degree is increased, K will converge to a specific value. However, the computations then would involve excessively high CPU times and internal memory capacity and thus are not executed.

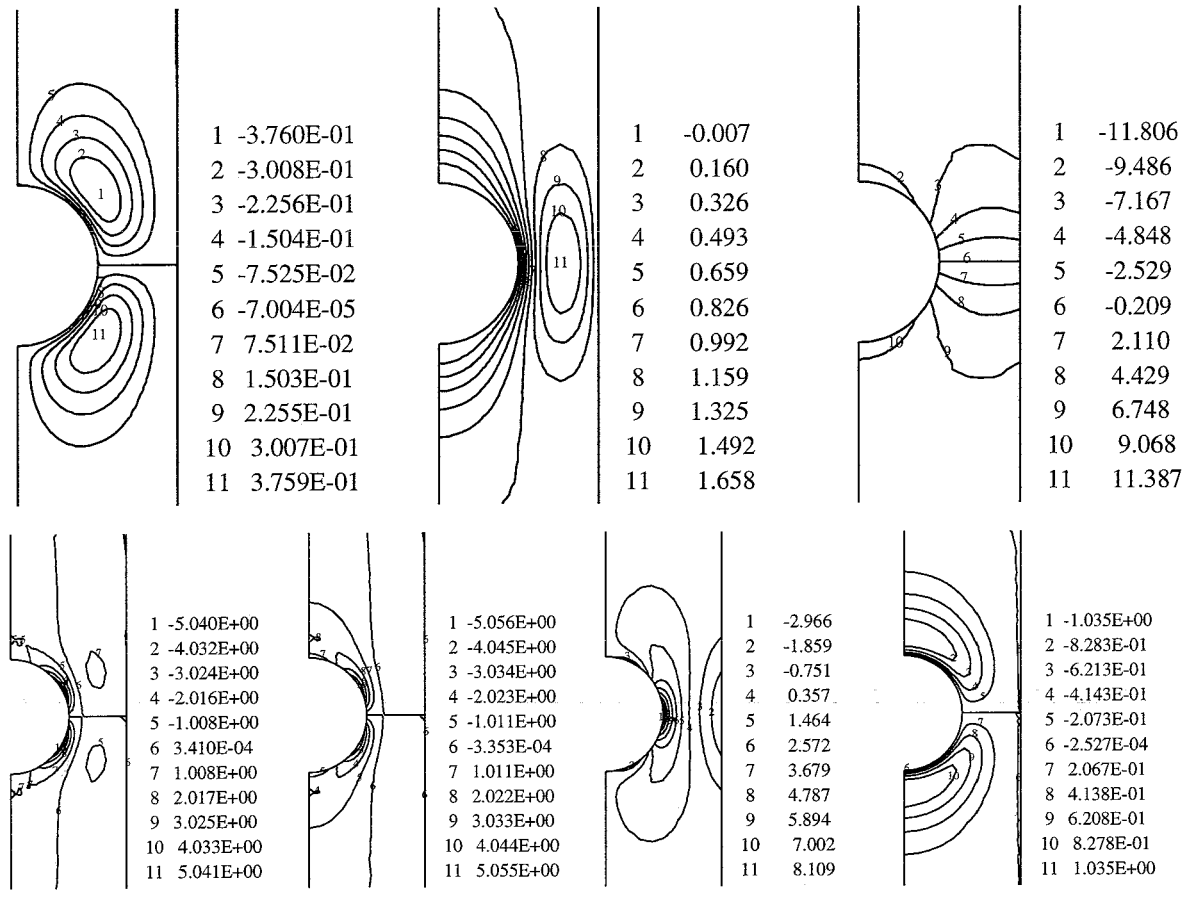

Figure 4.7: *Contourplots of a Newtonian flow (We = 0.0) for N = 2; top: from left to right: u_r, u_z, p; bottom: from left to right: τ_{rr}, τ_{zz}, τ_{rz}, $\tau_{\theta\theta}$.*

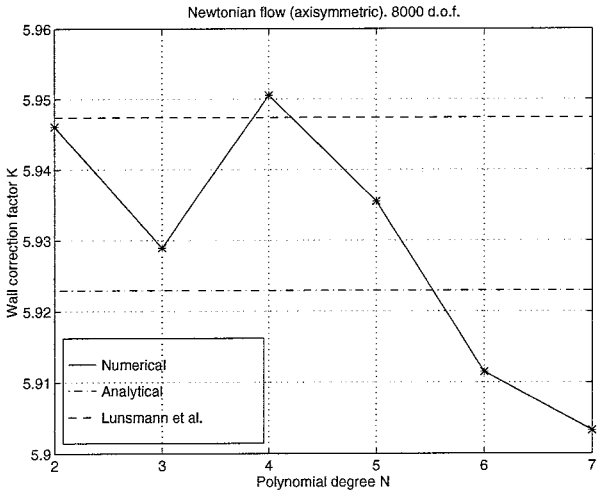

Figure 4.8: *Wall correction factor as a function of the polynomial degree N for a fixed number of degrees of freedom.*

4.2.2 Viscoelastic flow

For the wall correction factor K of viscoelastic flows as a function of the Weissenberg number We, a comparison is made between the different meshes M1 to M6 and the values from the literature (Lunsmann *et al.* [20] and Selen [24]). Table 4.3 and figure 4.9 present, respectively, the numerical values and the graphical representation.

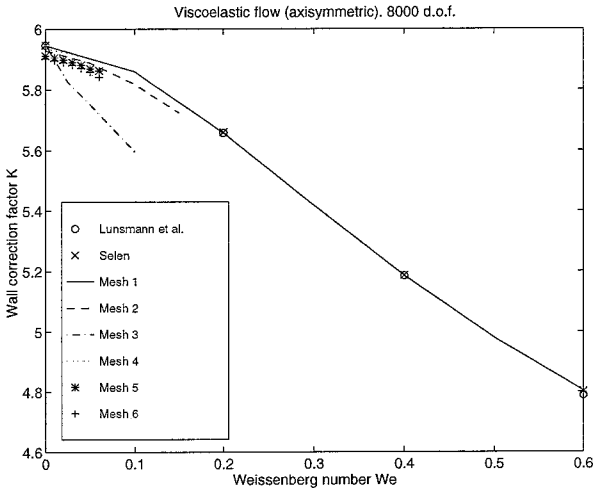

Figure 4.9: *Graph of the wall correction factor K as a function of the Weissenberg number.*

We	Lunsmann *et al.*	Selen	M1	M2	M3	M4	M5	M6
0.000	5.947	5.947	5.946	5.929	5.951	5.936	5.911	5.903
0.010							5.906	5.897
0.020							5.899	5.890
0.025					5.826	5.921		
0.030							5.891	5.882
0.040							5.881	5.871
0.050				5.889	5.751	5.904	5.869	5.858
0.060						diverges	5.863	5.842
0.075					5.673		diverges	diverges
0.100			5.859	5.818	5.595			
0.150				5.724	diverges			
0.200	5.658	5.660	5.657	diverges				
0.300			5.418					
0.400	5.185	5.186	5.186					
0.500			4.979					
0.600	4.788	4.801	4.803					
0.800	4.524	4.528	diverges					

Table 4.3: *Wall correction factor as a function of the Weissenberg number.*

For all the meshes, except mesh M3, the wall correction factors are in good agreement with the results from the literature. Remark also, that a loss of convergence at lower Weissenberg number We occurs as the polynomial degree N is increased. An explanation will be given in the next section.

4.2.3 Discussion

The strange results of mesh M3 for the viscoelastic wall correction factor are attributed to the inadequate resolution for this order N. In figure 4.10, τ_{rr}, τ_{zz} and τ_{rz} are plotted over the surface of the sphere for a Weissenberg number $We = 0.1$ and polynomial degree $N = 4$. Heavy oscillations are detected, where as the same plots for $N = 2$, shown in figure 4.11, are smooth. The resolution is not adequate enough and the results obtained with M3 are unreliable.

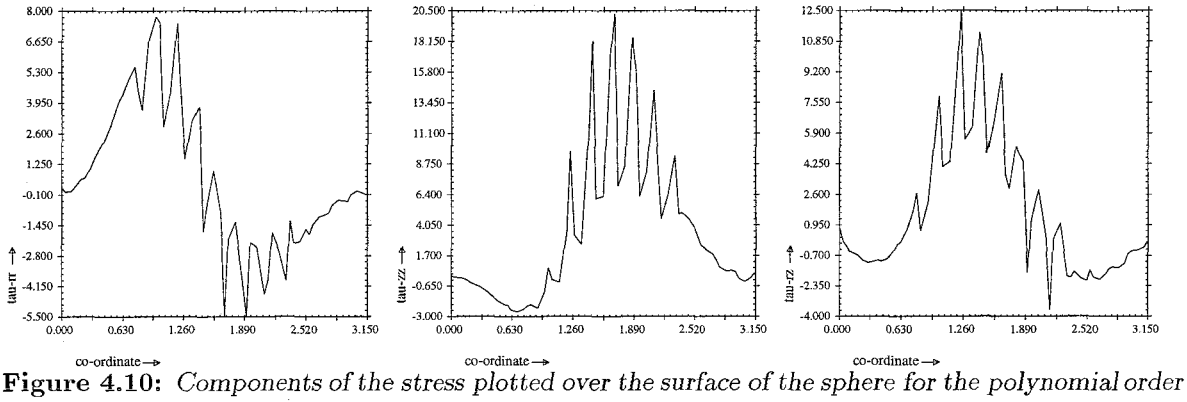

Figure 4.10: *Components of the stress plotted over the surface of the sphere for the polynomial order $N = 4$ for $We = 0.1$; from left to right: τ_{rr}, τ_{zz}, τ_{rz}.*

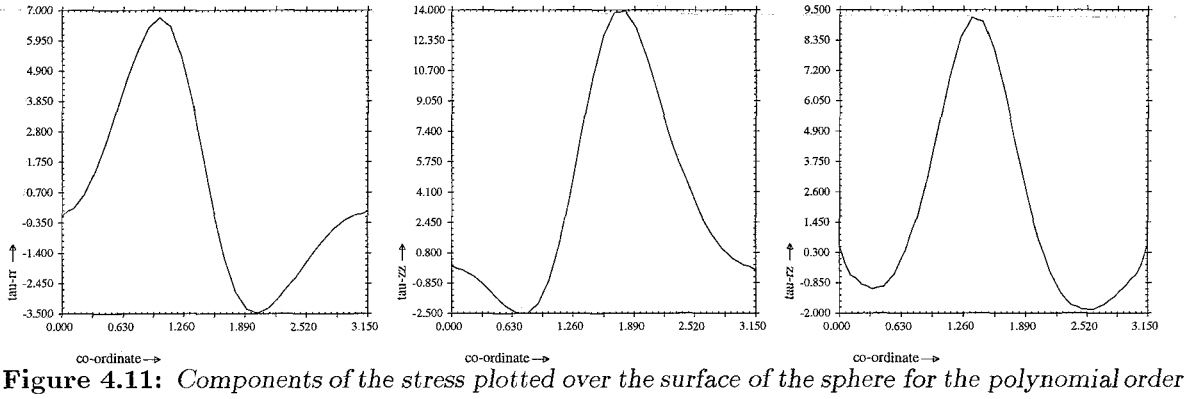

Figure 4.11: *Components of the stress plotted over the surface of the sphere for the polynomial order $N = 2$ for $We = 0.1$; from left to right: τ_{rr}, τ_{zz}, τ_{rz}.*

An additional reason exists, which also explains the divergence of the problem at lower We for higher N. The ill-conditioning of the large system matrix $\underline{A}$ of the linear set of equations $\underline{A}\,\underset{\sim}{x} = \underset{\sim}{b}$ (in our case set (3.47)) leads to large errors in the solution $\underset{\sim}{x}$ as well. An estimation of the relative error in the solution can be given as:

$$\frac{\| \bar{\underset{\sim}{x}} - \underset{\sim}{x} \|}{\| \underset{\sim}{x} \|} \leq c(\underline{A}) \cdot \eta \,, \tag{4.10}$$

where $\bar{\underset{\sim}{x}}$ is the approximate solution, $\underset{\sim}{x}$ the exact solution, $c(\underline{A})$ is the condition number of the system matrix $\underline{A}$, and η the machine accuracy. The condition number $c(\underline{A})$ denotes the sensibility of the solution for perturbations in the matrix $\underline{A}$ and is given as:

$$c(\underline{A}) = \| \underline{A} \| \cdot \| \underline{A}^{-1} \| \qquad \text{with } \| \underline{A} \| = \max\left(\sqrt{\sigma(\underline{A}^T \underline{A})} \right) . \tag{4.11}$$

$\sigma(\underline{A}^T \underline{A})$ is the spectrum of eigenvalues of the matrix $\underline{A}^T \underline{A}$.

Especially for high order methods, the system matrix is ill-conditioned (Canuto *et al.* [17]). The approximate solution is used to build the new large system matrix $\underline{A}$ and right hand side $\underline{b}$. Thus, errors in the solution are transmitted to the system matrix and right hand side, possibly leading to accumulation of errors and divergence of the problem.

The ill-conditioning of the matrix for higher orders could find its origin in the derivatives of the basis functions. With higher order elements, the derivatives of the basis functions have both higher and lower values. This is clearly visible in figure 4.12. For $N = 2$, near the interpolation points, the interpolants through the Legendre-Gauss-Lobatto points are not as steep as for $N = 6$. Especially at the boundary of the elements, the functions are very steep. At the same time, further away from the interpolation points, for $N = 6$ the interpolants through the LGL points are flatter than for $N = 2$.

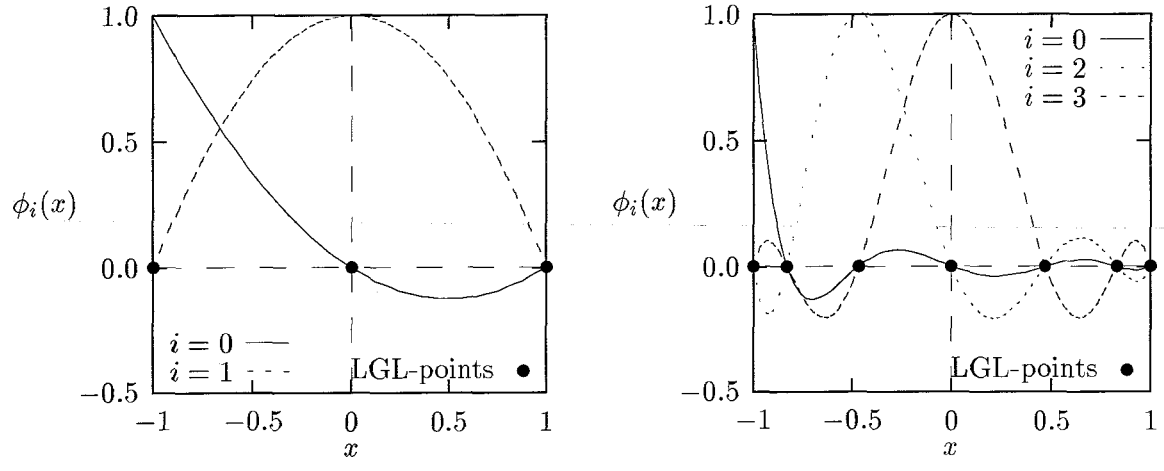

Figure 4.12: *Lagrange interpolants $\phi_i(x)(i = 0, \ldots, N)$ through the Legendre-Gauss-Lobatto points (•) for $N = 2$ (left) and $N = 6$ (right).*

The penalty parameter (Eqn. (3.48)), however, has influence on the matrix condition number as well. To obtain an idea of the influence of the order N and the penalty parameter on the condition of the system matrix, the relative sizes of the submatrices within the system matrix are examined. The dimensionless velocity $\vec{u}$, the test functions $\boldsymbol{S}_i^d$, $\vec{v}$, q and $\boldsymbol{G}$ and the dimensionless viscosities η_s and η_i have all maximum values approximately equal to unity. The maximum value of the gradient operator $\vec{\nabla}$ grows as the order N is increased. And the extra stress τ_i increases as the elasticity becomes more important. If the maximum values of the gradient operator, the relaxation time, the extra stress and the penalty parameter are denoted by, respectively, ∇, λ, τ and κ, the relative sizes of the different submatrices then read:

matrix	size	matrix	size	matrix	size	matrix	size
$\delta_\tau \underline{F}_i$	$\lambda\nabla + 1$	$\delta_\tau \underline{G}$	∇	$\delta_\tau \underline{E}$	0	$\delta_\tau \underline{H}$	0
$\delta_u \underline{F}_i$	$\lambda\tau\nabla + \nabla + \lambda\tau$	$\delta_u \underline{G}$	∇^2	$\delta_u \underline{E}$	∇	$\delta_u \underline{H}$	∇
$\delta_{\bar{D}} \underline{F}_i$	0	$\delta_{\bar{D}} \underline{G}$	∇	$\delta_{\bar{D}} \underline{E}$	1	$\delta_{\bar{D}} \underline{H}$	0
$\delta_p \underline{F}_i$	0	$\delta_p \underline{G}$	∇	$\delta_p \underline{E}$	0	$\delta_p \underline{H}$	κ

Table 4.4: *Relative sizes of the submatrices in the system matrix.*

If $\lambda\,\tau \approx 1$ is chosen, the relative sizes within the system matrix of set (3.47) is then approximately given by:

$$\underline{A}_1 = \begin{bmatrix} \nabla^2 + \nabla + \frac{1}{\kappa}\nabla^2 & \nabla \\ \nabla & 1 \end{bmatrix} . \tag{4.12}$$

In MATLAB, a 4×4 matrix is created by multiplying $\nabla^2 + \nabla + \frac{1}{\kappa}\nabla^2$ and ∇ with a 2×2 uniform distributed random matrix ranging between 0 and 1. For the bottom right submatrix $\delta_{\bar{D}}\underline{E}$, a 2×2 identity matrix is taken. The condition number of this matrix $\underline{A}_1$ is calculated for the ranges $10^{-10} < \kappa < 10^{-2}$ and $10 < \nabla < 100$. The results are given in figure 4.13. The assumption is made, that the same sort of influence of the order N and the penalty parameter κ on the condition of this matrix $\underline{A}_1$ also holds for the condition of the actual system matrix.

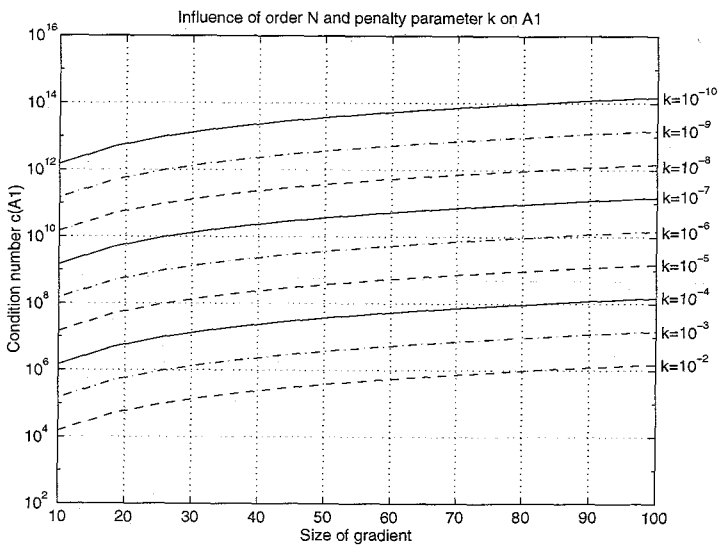

Figure 4.13: *The matrix condition number $c(\underline{A}_1)$ as a function of the penalty parameter κ and order N.*

The penalty parameter has a disastrous influence on the condition. A better method is to directly solve the pressure instead of eliminating it at element level. The system then reads:

$$\begin{bmatrix} \delta_u\underline{G} - \delta_\tau\underline{G} \cdot (\delta_\tau\underline{F}_i)^{-1} \cdot \delta_u\underline{F}_i & \delta_{\bar{D}}\underline{G} & \delta_p\underline{G} \\ \delta_u\underline{E} & \delta_{\bar{D}}\underline{E} & \underline{0} \\ \delta_u\underline{H} & \underline{0} & \delta_p\underline{H} \end{bmatrix} \begin{bmatrix} \delta\underset{\sim}{u} \\ \delta\underset{\sim}{\bar{D}} \\ \delta\underset{\sim}{p} \end{bmatrix} = \begin{bmatrix} -\underset{\sim}{G} + \delta_\tau\underline{G} \cdot (\delta_\tau\underline{F}_i)^{-1} \cdot \underset{\sim}{F}_i \\ -\underset{\sim}{E} \\ -\underset{\sim}{H} \end{bmatrix} . \tag{4.13}$$

The relative sizes within the system matrix are then denoted by:

$$\underline{A}_2 = \begin{bmatrix} \nabla^2 + \nabla & \nabla & \nabla \\ \nabla & 1 & 0 \\ \nabla & 0 & \kappa \end{bmatrix} . \tag{4.14}$$

A 6×6 matrix is created in MATLAB in the same way as matrix $\underline{A}_1$. The penalty parameter κ is multiplied with a 2×2 identity matrix. The matrix condition number of matrix $\underline{A}_2$

as a function of the penalty parameter and the order N for this method is given in figure 4.14. Notice, that the penalty parameter has no influence at all on the condition number. Here, κ can possibly be chosen equal to zero and thus the actual problem is left without the compressibility.
At high polynomial orders N, the maximum size of the contribution of the discontinuous extra stress variable to the momentum equation is negligible to the maximum size of the velocity variable. Thus, the extra stress may not have any influence on the momentum equation and a different problem is solved. A continuous discretization of the extra stress would then be a better choice.

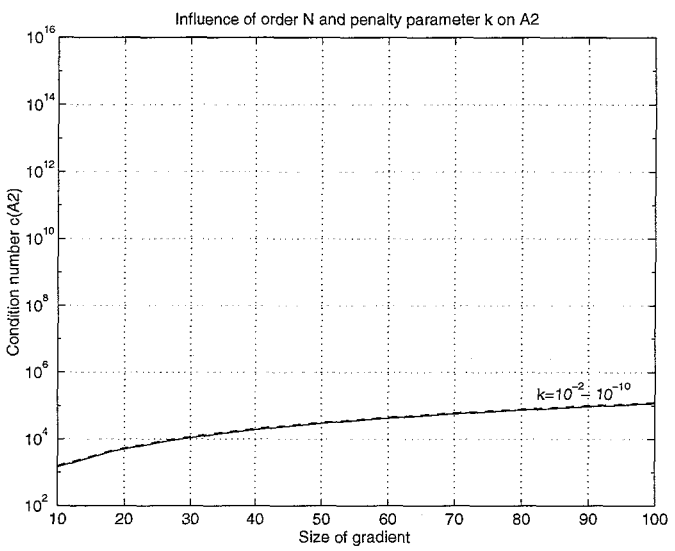

Figure 4.14: *The matrix condition number $c(\underline{A}_2)$ as a function of the penalty parameter κ and order N by directly solving the pressure.*

In our computations, a direct solver is used to solve the system of equations (3.47). It is based on the LU-decomposition without partial pivoting. This avoids the destruction of the band structure of the total system matrix. A disadvantage is that a penalty parameter is necessary to avoid zero pivots on the diagonal. However, a direct solver is not an efficient way of handling the method, especially not for higher polynomial order. Increasing the order N means more nodes per element and thus enlarging the system matrix on element level. A result is a broader band width in the total system. As less elements are needed for higher order methods, the large system matrix has more non-zero elements. So, to solve the set of equations more efficiently and to gain computing time, an iterative solver is required. Especially on element level, where a full system matrix is present, an iterative solver is more appropriate. At the moment, a LU-decomposition with partial pivoting solves the stress and pressure variables on element level.

In order to obtain reasonable convergence of the iterative solver, an efficient preconditioner is needed due to the ill-conditioning of the system matrix. For higher order methods, Deville and Mund [35] proposed a suitable preconditioner, which uses the system matrix of a linear finite element approximation with its nodes corresponding with the nodes of the spectral

element system (see figure 4.15). A basic iterative scheme to solve $\underline{A}\,\underset{\sim}{x} = \underset{\sim}{b}$ is then given by:

$$\underline{F}\,\underset{\sim}{x}^0 = \underset{\sim}{b} \tag{4.15}$$

$$\underline{F}\,\underset{\sim}{x}^k = \underline{F}\,\underset{\sim}{x}^{k-1} + \alpha(\underset{\sim}{b} - \underline{A}\,\underset{\sim}{x}^{k-1}) \,, \tag{4.16}$$

where $\underline{F}$ denotes the finite element matrix, $\underset{\sim}{x}$ the solution vector, $\underset{\sim}{b}$ the right hand side vector, α a dimensionless parameter and $\underline{A}$ the spectral system matrix. Note that only low order finite element systems have to be solved and the spectral system $\underline{A}$ is only used to compute the spectral residual. The convergence of the finite element preconditioned iterative solver is very fast and only a few iterations are needed to decrease the error to machine accuracy (Timmermans and Van de Vosse [36]).

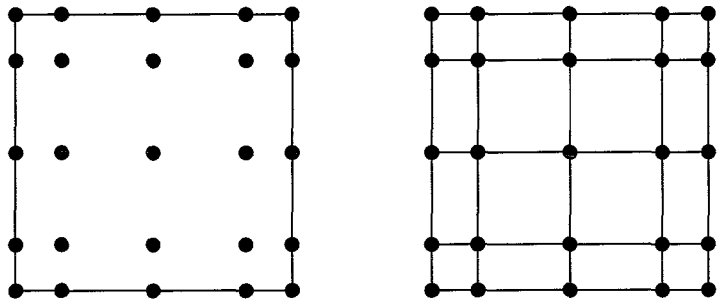

Figure 4.15: *Spectral and corresponding finite element mesh.*

Although an iterative solver handles the numerical calculations more efficiently, it does not improve the condition of the actual problem. So, the large errors in the solution still arise for small errors in the right hand side. In other words, divergence is still very likely to occur already at low Weissenberg numbers *We* for high order approximation of the flow. As shown above, directly solving the pressure improves the condition and leaves the possibility to omit the penalty parameter. In future, further research is needed to overcome the ill-conditioning of the higher order Stabilized Discontinuous Galerkin method. As well, the negligibility of the discontinuous extra stress variable in the momentum equation for higher orders is not clear yet and needs more investigation. Possibly, a continuous discretization for higher orders is advised. An extra disadvantage is the increase in computing time, especially for a multi-mode approach.

Chapter 5

Conclusions and recommendations

This report presents a spectral element method to analyze numerically a viscoelastic flow around a falling sphere in a tube. The method combines the flexibility of the classical finite element method and the accuracy of spectral methods; within each element, the solution is approximated using high order polynomials. A Stabilized Discontinuous Galerkin approach together with a Upper Convected Maxwell constitutive model is used to describe the flow problem. By eliminating the extra stress variable at element level, a discontinuous discretization of this variable is accomplished and an efficient handling for multi-mode models is the result. The discrete rate of deformation is added as an extra variable to the set of equations. This methodology leaves a more stable numerical method. It improves the compatibility between the velocity and the stress (Selen [24]). Further, the discretization of our extra stress - velocity - discrete rate of deformation - pressure formulation is chosen to be $(\boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p) \mapsto (Q^d_{N-1,K}, Q_{N,K}, Q_{N,K}, Q^d_{N-2,K})$. For the numerical integration, a high order Legendre-Gauss quadrature rule is applied. The linearized the set of equations is solved by Newton iteration using a direct solver based on the LU-decomposition without partial pivoting.

For a simple Poisseuille flow under Stokes' conditions, the method shows an algebraic rate of convergence by augmenting the number of elements. If the polynomial order of the approximation functions is increased, an exponential convergence is detected. For calculations with approximately the same number of degrees of freedom, but different polynomial orders, higher order approximation shows more accurate results.

The benchmark problem of the steady flow past a falling sphere in a tube is much more complex. The calculations are difficult, because of the presence of thin stress boundary layers in the region near the surface of the sphere and a strong elongational flow behind the sphere. Numerical results for this benchmark problem are obtained with different meshes. The number of degrees of freedom are approximately equal for all meshes, but the polynomial orders and therefore the distribution of the nodes are different. The calculations are in good agreement with results found in the literature, except for one mesh. Results with this mesh M3 show an inadequate resolution (see the figures 4.6, 4.9 and 4.10). For the Newtonian flow, the numerical wall correction factor does not converge to a certain value if the order is increased. Also, for the viscoelastic calculations, loss of convergence occurs at lower Weissenberg number We for higher polynomial degree N. This is in contradiction of what was expected from

the theory: higher order approximation has a higher accuracy and thus higher Weissenberg numbers were expected to be reached. The mesh is believed to be insufficiently fine and therefore gives rise to errors in the solution. However, computations with finer meshes were not executed, because that would involve excessively high CPU times and internal memory capacity.
In combination with an insufficient discretization, the ill-conditioning of the large system matrix results in divergence of the problem at lower We for higher orders. The solution is used to build the new large system matrix and right hand side. So, errors in the solution are reproduced in the matrix and right hand side as well. The ill-conditioning magnifies the errors, leading to divergence. As increasing the polynomial order N means a more ill-conditioned system, a loss of convergence at lower Weissenberg numbers is the result.

In our method, the pressure variable is eliminated at element level. This involves a penalty parameter, which deteriorates the condition of the large system matrix. By directly solving the pressure, considerable improvement of the condition number is obtained. Further investigation on the condition number of the large system matrix and its improvement is advised, especially for higher order methods.

For faster implementation of the Spectral Element Method, a program based on the Finite Element Method was adjusted. This program applies a Gauss quadrature rule and is inefficiently programmed for adjustment to higher order approximation. Improvement of the program and a Legendre-Gauss-Lobatto quadrature rule would enormously reduce the computational time and data storage. Then, computations with finer meshes are able to be executed and higher We may be obtained. Because higher order methods have more nodes per element and less elements are needed, the large system matrix has a broader band width and more non-zero elements. Therefore, a preconditioned iterative solver handles the problem more efficiently instead of the direct solver based on the LU-decomposition.

For higher polynomial orders, the elimination of the extra stress at element level in the momentum equation may be negligible compared to other terms. If so, a continuous approximation of the extra stress variable is preferred. However, this is not clear yet and needs further research.

The method used in this report, determines directly the steady state solution of the problem. However, the transitions in time of the flow before reaching the steady state solution may give a better insight in viscoelastic flows. As well, this would involve a possibility to calculate instationary flows. A method using a time discretization is the Pressure Correction Method. It tackles problems in a decoupled sense. A disadvantage of decoupled methods, as suggested by Owens [22], would seem to be that it is not possible in general to reach as high a limiting value of Weissenberg number as for corresponding coupled methods.

Bibliography

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

[2] R.C. King, M.R. Apelian, R.C. Armstrong, and R.A. Brown. Numerically stable finite element techniques for viscoelastic calculations in smooth and singular domains. *J. of Non-Newtonian Fluid Mech.*, 29:147–216, 1988.

[3] M.A. Mendelson, P.W. Yeh, R.C. Armstrong, and R.A. Brown. *J. of Non-Newtonian Fluid Mech.*, 10:31–54, 1982.

[4] 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. *J. of Non-Newtonian Fluid Mech.*, 36:159–192, 1990.

[5] A.N. Brooks and T.J.R. Hughes. Streamline Upwind / Petrov Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Comp. Meth. in Appl. Mech. and Eng.*, 32:199–259, 1982.

[6] P. Lesaint and P.A. Raviart. *On a finite element method for solving the neutron transport equation.* Academic Press, New York, 1974.

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

[8] L.A. Ying. *Computational Mechanics*, 2:45–53, 1987.

[9] M. Fortin and R. Pierre. On the convergence of the mixed method of Crochet and Marchal for viscoelastic flows. *Intern. J. Num. Methods Fluids*, 7:1035–1052, 1987.

[10] A. Fortin, A. Zine, and J-F. Agassant. Computing viscoelastic fluid flow problems at low cost. *J. of Non-Newtonian Fluid Mech.*, 45:209–229, 1992.

[11] F.P.T. Baaijens. Numerical analysis of start-up planar and axisymmetric contraction flows using multi-mode differential constitutive models. *J. of Non-Newtonian Fluid Mech.*, 48:147–180, 1993.

[12] A.N. Beris, R.C. Armstrong, and R.A. Brown. Spectral finite-element calculations for the flow of a Maxwell fluid. *J. of Non-Newtonian Fluid Mech.*, 22:129–167, 1987.

[13] R.G. Owens and T.N. Phillips. A spectral domain decomposition method for the planar non-Newtonian stick-slip problem. *J. of Non-Newtonian Fluid Mech.*, 41:43–79, 1991.

[14] A. Souvaliotis and A.N. Beris. Applications of domain decomposition spectral collocation methods in viscoelastic flows through model porous media. *J. Rheol.*, 36:1417–1453, 1992.

[15] V. Van Kemenade and M.O. Deville. Application of spectral elements to viscoelastic creeping flows. *J. of Non-Newtonian Fluid Mech.*, 51:277–308, 1994.

[16] K.K. Talwar and B. Khomami. Higher order finite element techniques for viscoelastic flow problems with change of type and singularities. *J. of Non-Newtonian Fluid Mech.*, 59:49–72, 1995.

[17] C. Canuto, M.Y. Hussaini, A. Quarteroni, and T. Zang. *Spectral Methods in Fluid Dynamics.* Springer Verlag, New York, 1988.

[18] M.D. Chilcott and J.M. Rallison. Creeping flow of dilute polymer solutions past cylinders and spheres. *J. of Non-Newtonian Fluid Mech.*, 29:381–432, 1988.

[19] O.G. Harlen. High-Deborah number flow of a dilute polymer solution past a sphere falling along the axis of a cylindrical tube. *J. of Non-Newtonian Fluid Mech.*, 37:157–173, 1990.

[20] W.J. Lunsmann, L. Genieser, R.C. Armstrong, and R.A. Brown. Finite element analysis of steady viscoelastic flow around a sphere in a tube: calculations with constant viscosity models. *J. of Non-Newtonian Fluid Mech.*, 48:63–99, 1993.

[21] Fan Yurun and M.J. Crochet. High-order finite element methods for steady viscoelastic flows. *J. of Non-Newtonian Fluid Mech.*, 57:283–311, 1995.

[22] R.G. Owens. Steady viscoelastic flow past a sphere using spectral elements. *Int. J. Num. Meth. Eng.*, 39:1517–1534, 1996.

[23] M.J. Crochet, A.R. Davies, and K. Walters. *Numerical Simulation of Non-Newtonian Flow*, volume Rheology Series, 1. Elsevier Science Publishers B.V., Amsterdam, 1984.

[24] J.H.A. Selen. Multi-mode viscoelastic computations of complex flows. Master's thesis, Stan Ackermans Institute, Eindhoven University of Technology, WFW 95.141, 1995.

[25] R.B. Bird, R.C. Armstrong, and O. Hassager. *Dynamics of Polymeric Liquids*, volume 1. Fluid Mechanics. John Wiley & Sons, New York, 1987.

[26] R.G. Larson. *Constitutive Equations for Polymer Melts and Solutions.* Butterworths, Boston, 1988.

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

[28] F.P.T. Baaijens. Applied Computational Mechanics 2, pages 20-22, 1992. Eindhoven University of Technology.

[29] F.P.T. Baaijens. Numerical experiments with a Discontinuous Galerkin method including monotonicity enforcement on the stick-slip problem. *J. of Non-Newtonian Fluid Mech.*, 51:141–159, 1994.

[30] F.P.T. Baaijens. Application of low-order Discontinuous Galerkin methods to the analysis of viscoelastic flows. *J. of Non-Newtonian Fluid Mech.*, 52:37–57, 1994.

[31] A.T. Patera. A spectral element method for fluid dynamics: laminar flow in a channel expansion. *J. Comput. Phys.*, 54:468–488, 1984.

[32] Y. Maday and A.T. Patera. Spectral element methods for the incompressible Navier-Stokes equations. *A.K. Noor and J.T. Oden, (Eds.) in State-of-the-Art Surveys on Computational Mechanics, ASME, New York*, pages 71–143, 1989.

[33] E.M. Rønquist. *Optimal spectral element methods for the unsteady three-dimensional incompressible Navier-Stokes equations.* PhD thesis, Massachusetts Institute of Technology, Cambridge, USA, 1988.

[34] J. Happel and H. Brenner. *Low Reynolds Number Hydrodynamics.* Martinus Nijhoff, New York, 1986.

[35] M.O. Deville and E. Mund. Finite element preconditioning for pseudospectral solutions of elliptic problems. *SIAM J. Sci. Stat. Comp.*, 11:311–342, 1985.

[36] L.J.P. Timmermans and F.N. van de Vosse. Finite element preconditioned spectral element methods for convection-diffusion problems. In F. Dijksman, editor, *Topics in applied mechanics: integration of theory and applications in mechanics.* Kluwer Academic Publishers, 1994.

Appendix A

Linearized matrices for the SDG method

The dimensionless mixed weak formulation of the Stabilized Discontinuous Galerkin method is given by:

Problem SDGE *Find* $(\boldsymbol{\tau}_i, \vec{u}, p, \bar{\boldsymbol{D}}) \in \mathcal{S} \times \mathcal{U} \times \mathcal{Q} \times \mathcal{G}$ *such that for all* $(\boldsymbol{S}_i^d, \vec{v}, q, \boldsymbol{G}) \in \mathcal{S} \times \mathcal{V} \times \mathcal{Q} \times \mathcal{G}$

$$(\boldsymbol{S}_i^d, We_i \mathcal{L}_i^* \boldsymbol{\tau}_i + \boldsymbol{Y}_i \cdot \boldsymbol{\tau}_i - 2\eta_i \boldsymbol{D}_u)$$

$$-\sum_{e=1}^{K} \left[\int_{\Gamma_{in}^e} \boldsymbol{S}_i^d : We_i \, \vec{u} \cdot \vec{n} (\boldsymbol{\tau}_i - \boldsymbol{\tau}_{in}^{ext}) d\Gamma \right] = 0 \qquad i = 1, \ldots, M \ , \tag{A.1}$$

$$-\Big(\frac{1}{Re}\boldsymbol{D}_v, 2\eta_s \boldsymbol{D}_u + 2\beta \sum_{i=1}^{M} \eta_i (\boldsymbol{D}_u - \bar{\boldsymbol{D}}) + \sum_{i=1}^{M} \boldsymbol{\tau}_i \Big) + (\frac{1}{Re} \vec{\nabla} \cdot \vec{v}, p) = 0 \ , \tag{A.2}$$

$$(\boldsymbol{G}, \bar{\boldsymbol{D}} - \boldsymbol{D}_u) = 0 \ , \tag{A.3}$$

$$(q, \vec{\nabla} \cdot \vec{u}) = 0 \ . \tag{A.4}$$

For implementation in the finite element program SEPRAN, this set of equations is discretized and linearized. Denote $\delta\boldsymbol{\tau}_i$, $\delta\vec{u}$, $\delta\bar{\boldsymbol{D}}$ and δp as the variations of the extra stress tensor of a mode, the velocity field, the discrete rate of deformation tensor and the pressure, respectively. And define the short notation for Eqn. (A.1) as $F_i(\boldsymbol{S}_i^d, \vec{v}, \boldsymbol{G}, q; \boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p)$, for Eqn. (A.2) as $G(\boldsymbol{S}_i^d, \vec{v}, \boldsymbol{G}, q; \boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p)$, for Eqn. (A.3) as $E(\boldsymbol{S}_i^d, \vec{v}, \boldsymbol{G}, q; \boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p)$ and for Eqn. (A.4) as $H(\boldsymbol{S}_i^d, \vec{v}, \boldsymbol{G}, q; \boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p)$. Assuming that the directional derivatives of the above mentioned equations exist, the definition for a derivative is given as:

$$\delta_u F_i(\boldsymbol{S}_i^d, \vec{v}, \boldsymbol{G}, q; \boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p; \delta\vec{u}) = \lim_{\theta \to 0} \frac{F_i(\boldsymbol{S}_i^d, \vec{v}, \boldsymbol{G}, q; \boldsymbol{\tau}_i, \vec{u} + \theta\delta\vec{u}, \bar{\boldsymbol{D}}, p) - F_i(\boldsymbol{S}_i^d, \vec{v}, \boldsymbol{G}, q; \boldsymbol{\tau}_i, \vec{u}, \bar{\boldsymbol{D}}, p)}{\theta} \ . \tag{A.5}$$

For the discretization, a short notation of:

$$\tau_i(\xi_{Gu},\eta_{Gu}) = \sum_{j=0}^{N-1}\sum_{k=0}^{N-1} \mu_{jk_{LGLt}}(\xi_{Gu},\eta_{Gu})\tau_i(\xi_{j_{LGLt}},\eta_{k_{LGLt}}) \tag{A.6}$$

will be defined as:

$$\tau_{i_{Gu}} = \mu_{Gu}\tau_{i_{LGLt}} \ , \tag{A.7}$$

where $\tau_i(\xi_{Gu},\eta_{Gu})$ is the value of τ_i in the global coordinates, corresponding to the local Legendre-Gauss points ξ_{Gu} and η_{Gu} of the same order as the velocity (N) in the interval $[-1,1]^2$. $\mu_{jk_{LGLt}}$ is the basis function through the Legendre-Gauss-Lobatto points corresponding to the $\big(j+(k-1)(N-1)\big)$th Legendre-Gauss-Lobatto stress point of the parent element $[-1,1]^2$. $\tau_i(\xi_{j_{LGLt}},\eta_{k_{LGLt}})$ denotes the value of τ_i in the global point, corresponding to the $\big(j+(k-1)(N-1)\big)$th local Legendre-Gauss-Lobatto coordinates $\xi_{j_{LGLt}}$ and $\eta_{k_{LGLt}}$ of the same order as the stress $(N-1)$ in the interval $[-1,1]^2$.
The discretized matrix formulation of the different variables and its trial functions per element now read:

$$\underset{\sim}{\tau}_{i_{Gu}} = \underline{\mu}_{Gu}\underset{\sim}{\tau}_{i_{LGLt}} \qquad \longrightarrow \underset{\sim}{S}^d_{i_{Gu}} = \underline{\mu}_{Gu}\underset{\sim}{S}^d_i \ , \tag{A.8}$$

$$\underset{\sim}{u}_{Gu} = \underline{\varphi}_{Gu}\underset{\sim}{u}_{LGLu} \qquad \longrightarrow \underset{\sim}{v}_{Gu} = \underline{\varphi}_{Gu}\underset{\sim}{v} \ , \tag{A.9}$$

$$\underset{\sim}{p}_{Gu} = \underline{\psi}_{Gu}\underset{\sim}{p}_{Gp} \qquad \longrightarrow \underset{\sim}{q}_{Gu} = \underline{\psi}_{Gu}\underset{\sim}{q} \ , \tag{A.10}$$

$$\underset{\sim}{\bar{D}}_{Gu} = \underline{\chi}_{Gu}\underset{\sim}{\bar{D}}_{LGLd} \qquad \longrightarrow \underset{\sim}{G}_{Gu} = \underline{\chi}_{Gu}\underset{\sim}{G} \ , \tag{A.11}$$

where the vector $\underset{\sim}{\tau}_{i_{Gu}}$ contains the values per mode i of all the stress components for a single element. The vector $\underset{\sim}{\tau}_{i_{LGLt}}$ contains the values per mode i of all the stress components in all the Legendre-Gauss-Lobatto nodes per element. $\underline{\mu}$, $\underline{\varphi}$, $\underline{\psi}$ and $\underline{\chi}$ are the matrices containing the values of the basis functions in the Legendre-Gauss integration points for respectively the stress, the velocity, the pressure and the discrete rate of deformation.

The inner product over an element of the constitutive equation without the boundary integral is written as:

$$K_i = \int_{\Omega^e}\Bigg\{\underbrace{\lambda_i \boldsymbol{S}_i^d : \left(\frac{\boldsymbol{\tau}_i - \boldsymbol{\tau}_i^n}{\Delta t}\right)}_{(1)} + \underbrace{\lambda_i \boldsymbol{S}_i^d : \vec{u}\cdot\vec{\nabla}\boldsymbol{\tau}_i}_{(2)} - \underbrace{\lambda_i \boldsymbol{S}_i^d : \left\{(\boldsymbol{L}-\xi_i\boldsymbol{D}_u)\cdot\boldsymbol{\tau}_i + \boldsymbol{\tau}_i\cdot(\boldsymbol{L}-\xi_i\boldsymbol{D}_u)^c\right\}}_{(3)}$$
$$+ \underbrace{\left(\frac{\alpha_i\lambda_i}{\eta_i}\right)\boldsymbol{S}_i^d : \boldsymbol{\tau}_i\cdot\boldsymbol{\tau}_i}_{(4)} + \underbrace{\left(1+\frac{\epsilon_i\lambda_i}{\eta_i}tr(\boldsymbol{\tau}_i)\right)\boldsymbol{S}_i^d : \boldsymbol{\tau}_i}_{(5)} - \underbrace{2\eta_i \boldsymbol{S}_i^d : \boldsymbol{D}_u}_{(6)}\Bigg\} d\Omega^e \ . \tag{A.12}$$

If $w_{j,Gu}$ is defined as the weight in the jth Legendre-Gauss integration point, the discretized matrix formulations of the different parts of the constitutive equation are given as:

(1a) $\int_{\Omega^e}\lambda_i \boldsymbol{S}_i^d : \frac{\boldsymbol{\tau}_i}{\Delta t}d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu}\underset{\sim}{S}^{d^T}_{i_{j,Gu}}\frac{\lambda_i}{\Delta t}\underline{L}_1\underset{\sim}{\tau}_{i_{j,Gu}}$ with $\underline{L}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$,

(1b) $-\int_{\Omega^e} \lambda_i \boldsymbol{S}_i^d : \frac{\boldsymbol{\tau}_i^n}{\Delta t} d\Omega^e \Rightarrow -\sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{S}_{i_{j,Gu}}^{d^T} \frac{\lambda_i}{\Delta t} \underline{L}_1 \underset{\sim}{\tau}_{i_{j,Gu}}^n$,

(2) $\int_{\Omega^e} \lambda_i \boldsymbol{S}_i^d : \vec{u} \cdot \vec{\nabla} \boldsymbol{\tau}_i d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{S}_{i_{j,Gu}}^{d^T} \lambda_i \underline{L}_{7_j} \underset{\sim}{\tau}_{i_{j,Gu}}$

$$\text{with } \underline{L}_{7_j} = \begin{bmatrix} u_1 \frac{\partial}{\partial r} + u_2 \frac{\partial}{\partial z} & 0 & 0 & 0 \\ 0 & u_1 \frac{\partial}{\partial r} + u_2 \frac{\partial}{\partial z} & 0 & 0 \\ 0 & 0 & 2u_1 \frac{\partial}{\partial r} + 2u_2 \frac{\partial}{\partial z} & 0 \\ 0 & 0 & 0 & u_1 \frac{\partial}{\partial r} + u_2 \frac{\partial}{\partial z} \end{bmatrix}_j,$$

(3) $\int_{\Omega^e} \left\{ -\lambda_i \boldsymbol{S}_i^d : (\boldsymbol{L} - \xi_i \boldsymbol{D}_u) \cdot \boldsymbol{\tau}_i - \lambda_i \boldsymbol{S}_i^d : \boldsymbol{\tau}_i \cdot (\boldsymbol{L} - \xi_i \boldsymbol{D}_u)^c \right\} d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{S}_{i_{j,Gu}}^{d^T} \lambda_i \underline{L}_{2_j} \underset{\sim}{\tau}_{i_{j,Gu}}$

$$\text{with } \underline{L}_{2_j} = \begin{bmatrix} 2(\xi - 1) \frac{\partial u_1}{\partial r} & 0 & (\xi_i - 2) \frac{\partial u_1}{\partial z} + \xi_i \frac{\partial u_2}{\partial r} & 0 \\ 0 & 2(\xi_i - 1) \frac{\partial u_2}{\partial z} & \xi_i \frac{\partial u_1}{\partial z} + (\xi_i - 2) \frac{\partial u_2}{\partial r} & 0 \\ \xi_i \frac{\partial u_1}{\partial z} + (\xi_i - 2) \frac{\partial u_2}{\partial r} & (\xi_i - 2) \frac{\partial u_1}{\partial z} + \xi_i \frac{\partial u_2}{\partial r} & 2(\xi_i - 1) \left(\frac{\partial u_1}{\partial r} + \frac{\partial u_2}{\partial z} \right) & 0 \\ 0 & 0 & 0 & 2(\xi_i - 1) \frac{u_1}{r} \end{bmatrix}_j,$$

(4) $\int_{\Omega^e} \left(\frac{\alpha_i \lambda_i}{\eta_i} \right) \boldsymbol{S}_i^d : \boldsymbol{\tau}_i \cdot \boldsymbol{\tau}_i d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{S}_{i_{j,Gu}}^{d^T} \left(\frac{\alpha_i \lambda_i}{\eta_i} \right) \underline{L}_{3_j} \underset{\sim}{\tau}_{i_{j,Gu}}$

$$\text{with } \underline{L}_{3_j} = \begin{bmatrix} \tau_{11} & 0 & \tau_{12} & 0 \\ 0 & \tau_{22} & \tau_{12} & 0 \\ \tau_{12} & \tau_{12} & \tau_{11} + \tau_{22} & 0 \\ 0 & 0 & 0 & \tau_{33} \end{bmatrix}_j,$$

(5) $\int_{\Omega^e} \left(1 + \frac{\epsilon_i \lambda_i}{\eta_i} tr(\boldsymbol{\tau}_i) \right) \boldsymbol{S}_i^d : \boldsymbol{\tau}_i d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{S}_{i_{j,Gu}}^{d^T} \left(1 + \frac{\epsilon_i \lambda_i}{\eta_i} tr(\boldsymbol{\tau}_i) \right) \underline{L}_1 \underset{\sim}{\tau}_{i_{j,Gu}}$,

(6) $-\int_{\Omega^e} 2\eta_i \boldsymbol{S}_i^d : \boldsymbol{D}_u d\Omega^e \Rightarrow -\sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{S}_{i_{j,Gu}}^{d^T} 2\eta_i \underline{B}_j \underset{\sim}{u}_{i_{j,Gu}}$ with $\underline{B}_j = \begin{bmatrix} \frac{\partial}{\partial r} & 0 \\ 0 & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial z} & \frac{\partial}{\partial r} \\ \frac{1}{r} & 0 \end{bmatrix}_j$.

The derivative of K_i towards the variables $\boldsymbol{\tau}_i$ is given by:

$$\begin{aligned} \delta_\tau K_i = \int_{\Omega^e} \Bigg\{ & \underbrace{\lambda_i \boldsymbol{S}_i^d : \frac{\delta \boldsymbol{\tau}}{\Delta t}}_{(1)} + \underbrace{\lambda_i \boldsymbol{S}_i^d : \vec{u} \cdot \vec{\nabla} \delta \boldsymbol{\tau}_i}_{(2)} - \underbrace{\lambda_i \boldsymbol{S}_i^d : \left\{ (\boldsymbol{L} - \xi_i \boldsymbol{D}_u) \cdot \delta \boldsymbol{\tau}_i + \delta \boldsymbol{\tau}_i \cdot (\boldsymbol{L} - \xi_i \boldsymbol{D}_u)^c \right\}}_{(3)} \\ & + \underbrace{\left(\frac{\alpha_i \lambda_i}{\eta_i} \right) \boldsymbol{S}_i^d : (\delta \boldsymbol{\tau}_i \cdot \boldsymbol{\tau}_i + \boldsymbol{\tau}_i \cdot \delta \boldsymbol{\tau}_i)}_{(4)} + \underbrace{\left(1 + \frac{\epsilon_i \lambda_i}{\eta_i} tr(\boldsymbol{\tau}_i) \right) \boldsymbol{S}_i^d : \delta \boldsymbol{\tau}_i}_{(5)} + \underbrace{\left(\frac{\epsilon_i \lambda_i}{\eta_i} \right) \boldsymbol{S}_i^d : \delta \left(tr(\boldsymbol{\tau}_i) \right) \boldsymbol{\tau}_i}_{(6)} \Bigg\} d\Omega^e \,. \end{aligned} \tag{A.13}$$

The discretized matrix formulations of the different parts are given as:

(1) $\int_{\Omega^e} \lambda_i \boldsymbol{S}_i^d : \frac{\delta \boldsymbol{\tau}}{\Delta t} d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{S}_{i_{j,Gu}}^{d^T} \frac{\lambda_i}{\Delta t} \underline{L}_1 \delta \underset{\sim}{\tau}_{i_{j,Gu}}$,

(2) $\int_{\Omega^e} \lambda_i \boldsymbol{S}_i^d : \vec{u} \cdot \vec{\nabla} \delta\boldsymbol{\tau}_i d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{S}_{i_{j,Gu}}^{d^T} \lambda_i \underline{L}_{7_j} \delta\underset{\sim}{\tau}_{i_{j,Gu}}$,

(3) $\int_{\Omega^e} \{-\lambda_i \boldsymbol{S}_i^d : (\boldsymbol{L} - \xi_i \boldsymbol{D}_u) \cdot \delta\boldsymbol{\tau}_i - \lambda_i \boldsymbol{S}_i^d : \delta\boldsymbol{\tau}_i \cdot (\boldsymbol{L} - \xi_i \boldsymbol{D}_u)^c\} d\Omega^e \Rightarrow -\sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{S}_{i_{j,Gu}}^{d^T} \lambda_i \underline{L}_{2_j} \delta\underset{\sim}{\tau}_{i_{j,Gu}}$,

(4) $\int_{\Omega^e} \left(\frac{\alpha_i \lambda_i}{\eta_i}\right) \boldsymbol{S}_i^d : (\delta\boldsymbol{\tau}_i \cdot \boldsymbol{\tau}_i + \boldsymbol{\tau}_i \cdot \delta\boldsymbol{\tau}_i) d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{S}_{i_{j,Gu}}^{d^T} 2 \left(\frac{\alpha_i \lambda_i}{\eta_i}\right) \underline{L}_{3_j} \delta\underset{\sim}{\tau}_{i_{j,Gu}}$,

(5) $\int_{\Omega^e} \left(1 + \frac{\epsilon_i \lambda_i}{\eta_i} tr(\boldsymbol{\tau}_i)\right) \boldsymbol{S}_i^d : \delta\boldsymbol{\tau}_i d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{S}_{i_{j,Gu}}^{d^T} \left(1 + \frac{\epsilon_i \lambda_i}{\eta_i} tr(\boldsymbol{\tau}_i)\right) \underline{L}_1 \delta\underset{\sim}{\tau}_{i_{j,Gu}}$,

(6) $\int_{\Omega^e} \left(\frac{\epsilon_i \lambda_i}{\eta_i}\right) \boldsymbol{S}_i^d : \delta\left(tr(\boldsymbol{\tau}_i)\right) \boldsymbol{\tau}_i d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{S}_{i_{j,Gu}}^{d^T} \left(\frac{\epsilon_i \lambda_i}{\eta_i}\right) \underline{L}_{6_{i,j}} \delta\underset{\sim}{\tau}_{i_{j,Gu}}$

$$\text{with } \underline{L}_{6_{i,j}} = \begin{bmatrix} \tau_{11} & \tau_{11} & 0 & \tau_{11} \\ \tau_{22} & \tau_{22} & 0 & \tau_{22} \\ 2\tau_{12} & 2\tau_{12} & 0 & 2\tau_{12} \\ \tau_{33} & \tau_{33} & 0 & \tau_{33} \end{bmatrix}_{i,j} .$$

The derivative of K_i towards the variable $\vec{u}$ is given by:

$$\delta_u K_i = \int_{\Omega^e} \Big\{ \underbrace{\lambda_i \boldsymbol{S}_i^d : \delta\vec{u} \cdot \vec{\nabla}\boldsymbol{\tau}_i}_{(1)} - \underbrace{\lambda_i \boldsymbol{S}_i^d : (\delta_u \boldsymbol{L} - \xi_i \delta_u \boldsymbol{D}_u) \cdot \boldsymbol{\tau}_i}_{(2)} - \underbrace{\lambda_i \boldsymbol{S}_i^d : \boldsymbol{\tau}_i \cdot (\delta_u \boldsymbol{L} - \xi_i \delta_u \boldsymbol{D}_u)^c}_{(3)} - \underbrace{2\eta_i \boldsymbol{S}_i^d : \delta_u \boldsymbol{D}_u}_{(4)} \Big\} d\Omega^e \ . \quad \text{(A.14)}$$

The discretized matrix formulations of the different parts are given as:

(1) $\int_{\Omega^e} \lambda_i \boldsymbol{S}_i^d : \delta\vec{u} \cdot \vec{\nabla}\boldsymbol{\tau}_i d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{S}_{i_{j,Gu}}^{d^T} \lambda_i \underline{L}_{8_{i,j}} \delta\underset{\sim}{u}_{j,Gu}$ with $\underline{L}_{8_{i,j}} = \begin{bmatrix} \frac{\partial\tau_{11}}{\partial r} & \frac{\partial\tau_{11}}{\partial z} \\ \frac{\partial\tau_{22}}{\partial r} & \frac{\partial\tau_{22}}{\partial z} \\ 2\frac{\partial\tau_{12}}{\partial r} & 2\frac{\partial\tau_{12}}{\partial z} \\ \frac{\partial\tau_{33}}{\partial r} & \frac{\partial\tau_{33}}{\partial z} \end{bmatrix}_{i,j}$,

(2)+(3) $\int_{\Omega^e} \{-\lambda_i \boldsymbol{S}_i^d : (\delta_u \boldsymbol{L} - \xi_i \delta_u \boldsymbol{D}_u) \cdot \boldsymbol{\tau}_i - \lambda_i \boldsymbol{S}_i^d : \boldsymbol{\tau}_i \cdot (\delta_u \boldsymbol{L} - \xi_i \delta_u \boldsymbol{D}_u)^c\} d\Omega^e$

$$\Rightarrow -\sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{S}_{i_{j,Gu}}^{d^T} \lambda_i \underline{L}_{5_{i,j}} \delta\underset{\sim}{u}_{j,Gu}$$

$$\text{with } \underline{L}_{5_{i,j}} = \begin{bmatrix} 2(\xi_i - 1)\tau_{11}\frac{\partial}{\partial r} + (\xi_i - 2)\tau_{12}\frac{\partial}{\partial z} & \xi_i \tau_{12}\frac{\partial}{\partial r} \\ \xi_i \tau_{12}\frac{\partial}{\partial z} & 2(\xi_i - 1)\tau_{22}\frac{\partial}{\partial z} + (\xi_i - 2)\tau_{12}\frac{\partial}{\partial r} \\ \xi_i \tau_{11}\frac{\partial}{\partial z} + (\xi_i - 2)\tau_{22}\frac{\partial}{\partial z} + 2(\xi_i - 1)\tau_{12}\frac{\partial}{\partial r} & (\xi_i - 2)\tau_{11}\frac{\partial}{\partial r} + \xi_i \tau_{22}\frac{\partial}{\partial r} + 2(\xi_i - 1)\tau_{12}\frac{\partial}{\partial z} \\ 2(\xi_i - 1)\tau_{33}\frac{1}{r} & 0 \end{bmatrix}_{i,j}$$

(4) $-\int_{\Omega^e} 2\eta_i \boldsymbol{S}_i^d : \delta_u \boldsymbol{D}_u d\Omega^e \Rightarrow -\sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{S}_{i_{j,Gu}}^{d^T} 2\eta_i \underline{B}_j \delta\underset{\sim}{u}_{j,Gu}$.

The last two derivatives of K_i towards the variables p and $\bar{\boldsymbol{D}}$ are equal to zero.

The integrals must hold for all admissible test functions $\underset{\sim}{S}_i^d$. Thus, in total, the discretized matrix formulations of the constitutive equation K_i and its derivatives $\delta_\tau K_i$, $\delta_u K_i$, $\delta_p K_i$ and $\delta_{\bar{D}} K_i$ can be written as:

$$\begin{aligned}\underset{\sim}{K}_i = \sum_{j=1}^{n_{intp}} w_{j,Gu} \underline{\mu}^T_{i_{j,Gu}} \Bigg\{ & \frac{\lambda_i}{\Delta t} \underline{L}_1 \underline{\mu}_{i_{j,Gu}} \left(\underset{\sim}{\tau}_{i_{LGLt}} - \underset{\sim}{\tau}^n_{i_{LGLt}} \right) \\ & + \lambda_i \underline{L}_{7_j} \underline{\mu}_{i_{j,Gu}} \underset{\sim}{\tau}_{i_{LGLt}} + \lambda_i \underline{L}_{2_j} \underline{\mu}_{i_{j,Gu}} \underset{\sim}{\tau}_{i_{LGLt}} + \left(\frac{\alpha_i \lambda_i}{\eta_i} \right) \underline{L}_{3_j} \underline{\mu}_{i_{j,Gu}} \underset{\sim}{\tau}_{i_{LGLt}} \\ & + \left(1 + \frac{\epsilon_i \lambda_i}{\eta_i} tr(\boldsymbol{\tau}_i) \right) \underline{L}_1 \underline{\mu}_{i_{j,Gu}} \underset{\sim}{\tau}_{i_{LGLt}} - 2\eta_i \underline{B}_j \underline{\varphi}_{j,Gu} \underset{\sim}{u}_{LGLu} \Bigg\} \,, \end{aligned} \tag{A.15}$$

$$\begin{aligned}\delta_\tau \underline{K}_i \; \delta \underset{\sim}{\tau}_i = \sum_{j=1}^{n_{intp}} w_{j,Gu} \underline{\mu}^T_{i_{j,Gu}} \Bigg\{ & \frac{\lambda_i}{\Delta t} \underline{L}_1 + \lambda_i \underline{L}_{7_j} + \lambda_i \underline{L}_{2_j} + 2 \left(\frac{\alpha_i \lambda_i}{\eta_i} \right) \underline{L}_{3_j} \\ & + \left(1 + \frac{\epsilon_i \lambda_i}{\eta_i} tr(\boldsymbol{\tau}_i) \right) \underline{L}_1 + \left(\frac{\epsilon_i \lambda_i}{\eta_i} \right) \underline{L}_{6_{i,j}} \Bigg\} \underline{\mu}_{i_{j,Gu}} \delta \underset{\sim}{\tau}_{i_{LGLt}} \,, \end{aligned} \tag{A.16}$$

$$\delta_u \underline{K}_i \; \delta \underset{\sim}{u} = \sum_{j=1}^{n_{intp}} w_{j,Gu} \underline{\mu}^T_{i_{j,Gu}} \left\{ \lambda_i \underline{L}_{8_{i,j}} + \lambda_i \underline{L}_{5_{i,j}} + 2\eta_i \underline{B}_j \right\} \underline{\varphi}_{j,Gu} \delta \underset{\sim}{u}_{LGLu} \,, \tag{A.17}$$

$$\delta_p \underline{K}_i \; \delta \underset{\sim}{p} = \underline{0} \,, \tag{A.18}$$

$$\delta_{\bar{D}} \underline{K}_i \; \delta \underset{\sim}{\bar{D}} = \underline{0} \,. \tag{A.19}$$

The boundary integral in the constitutive equation and its derivatives can be written as:

$$L_i = \sum_{e=1}^{K} \int_{\Gamma_{in}^e} \boldsymbol{S}_i^d : \vec{u} \cdot \vec{n} \lambda_i (\boldsymbol{\tau}_i - \boldsymbol{\tau}_i^{n^{ext}}) d\Gamma \ , \tag{A.20}$$

$$\delta_\tau L_i = \sum_{e=1}^{K} \int_{\Gamma_{in}^e} \boldsymbol{S}_i^d : \vec{u} \cdot \vec{n} \lambda_i \delta\boldsymbol{\tau}_i d\Gamma \ , \tag{A.21}$$

$$\delta_u L_i = \sum_{e=1}^{K} \int_{\Gamma_{in}^e} \boldsymbol{S}_i^d : \delta\vec{u} \cdot \vec{n} \lambda_i (\boldsymbol{\tau}_i - \boldsymbol{\tau}_i^{n^{ext}}) d\Gamma \ . \tag{A.22}$$

The derivatives towards p and $\bar{\boldsymbol{D}}$ are zero again.

The integrals must hold for all admissible test functions $\underset{\sim}{S}_i^d$. So, the discretized matrix formulations give:

$$\underline{L}_i = \sum_{j=1}^{n_{intp}} w_{j,Gu} \underline{\mu}_{i_{j,Gu}}^T \underset{\sim}{n}^T \underline{\varphi}_{j,Gu} \underset{\sim}{u}_{LGLu} \lambda_i \underline{L}_1 \underline{\mu}_{i_{j,Gu}} \left(\underset{\sim}{\tau}_{i_{LGLt}} - \underset{\sim}{\tau}_{i_{LGLt}}^{n^{ext}} \right) \ , \tag{A.23}$$

$$\delta_\tau \underline{L}_i \ \delta\underset{\sim}{\tau}_i = \sum_{j=1}^{n_{intp}} w_{j,Gu} \underline{\mu}_{i_{j,Gu}}^T \lambda_i \underset{\sim}{n}^T \underline{\varphi}_{j,Gu} \underset{\sim}{u}_{LGLu} \underline{L}_1 \underline{\mu}_{i_{j,Gu}} \delta\underset{\sim}{\tau}_{i_{LGLt}} \ , \tag{A.24}$$

$$\delta_u \underline{L}_i \ \delta\underset{\sim}{u} = \sum_{j=1}^{n_{intp}} w_{j,Gu} \underline{\mu}_{i_{j,Gu}}^T \lambda_i \underline{L}_1 \underline{\mu}_{i_{j,Gu}} \left(\underset{\sim}{\tau}_{i_{LGLt}} - \underset{\sim}{\tau}_{i_{LGLt}}^{n^{ext}} \right) \underset{\sim}{n}^T \underline{\varphi}_{j,Gu} \delta\underset{\sim}{u}_{LGLu} \ , \tag{A.25}$$

$$\delta_p \underline{L}_i \ \delta\underset{\sim}{p} = \underline{0} \ , \tag{A.26}$$

$$\delta_{\bar{D}} \underline{L}_i \ \delta\underset{\sim}{\bar{D}} = \underline{0} \ . \tag{A.27}$$

The complete discretized matrix formulations of the constitutive equation and its derivatives then give:

$$\underline{F}_i = \underline{K}_i - \underline{L}_i \ , \tag{A.28}$$

$$\delta_\tau \underline{F}_i \ \delta\underset{\sim}{\tau}_i = (\delta_\tau \underline{K}_i - \delta_\tau \underline{L}_i) \ \delta\underset{\sim}{\tau}_i \ , \tag{A.29}$$

$$\delta_u \underline{F}_i \ \delta\underset{\sim}{u} = (\delta_u \underline{K}_i - \delta_u \underline{L}_i) \ \delta\underset{\sim}{u} \ , \tag{A.30}$$

$$\delta_p \underline{F}_i \ \delta\underset{\sim}{p} = (\delta_p \underline{K}_i - \delta_p \underline{L}_i) \ \delta\underset{\sim}{p} = \underline{0} \ , \tag{A.31}$$

$$\delta_{\bar{D}} \underline{F}_i \ \delta\underset{\sim}{\bar{D}} = (\delta_{\bar{D}} \underline{K}_i - \delta_{\bar{D}} \underline{L}_i) \ \delta\underset{\sim}{\bar{D}} = \underline{0} \ . \tag{A.32}$$

The inner product over an element of the equation of conservation of momentum is written as:

$$G = \int_{\Omega^e} \Big\{ \underbrace{- \boldsymbol{D}_v : \left(2\eta_0 + 2\beta \sum_{i=1}^{N} \eta_i \right) \boldsymbol{D}_u}_{(1)} + \underbrace{\boldsymbol{D}_v : \left(2\beta \sum_{i=1}^{N} \eta_i \right) \bar{\boldsymbol{D}}}_{(2)} \underbrace{- \boldsymbol{D}_v : \sum_{i=1}^{N} \boldsymbol{\tau}_i}_{(3)} + \underbrace{\vec{\nabla} \cdot \vec{v} p}_{(4)} \underbrace{- \vec{v} \cdot \rho \vec{u} \cdot (\vec{\nabla} \vec{u})}_{(5)} \Big\} d\Omega^e \ . \quad \text{(A.33)}$$

The discretized matrix formulations of the different parts are given as:

(1) $- \int_{\Omega^e} \boldsymbol{D}_v : (2\eta_0 + 2\beta \sum \eta_i) \, \boldsymbol{D}_u d\Omega^e \Rightarrow - \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{v}_{j,Gu}^T \, (2\eta_0 + 2\beta \sum \eta_i) \, \underline{B}_j^T \underline{L}_{20} \underline{B}_j \underset{\sim}{u}_{j,Gu}$

$$\text{with } \underline{L}_{20} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

(2) $\int_{\Omega^e} \boldsymbol{D}_v : (2\beta \sum \eta_i) \bar{\boldsymbol{D}} d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{v}_{j,Gu}^T (2\beta \sum \eta_i) \underline{B}_j^T \underline{L}_{30} \underset{\sim}{\bar{D}}_{j,Gu}$

$$\text{with } \underline{L}_{30} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{2} \\ -1 & -1 & 0 \end{bmatrix},$$

(3) $- \int_{\Omega^e} \boldsymbol{D}_v : \sum \boldsymbol{\tau}_i d\Omega^e \Rightarrow - \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{v}_{j,Gu}^T \underline{B}_j^T \sum \underset{\sim}{\tau}_{i_{j,Gu}}$,

(4) $\int_{\Omega^e} \vec{\nabla} \cdot \vec{v} p d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{v}_{j,Gu}^T \underline{V}_j^T \underset{\sim}{p}_{j,Gu}$ with $\underline{V}_j = \begin{bmatrix} \frac{\partial}{\partial r} + \frac{1}{r} & \frac{\partial}{\partial z} \end{bmatrix}_j$,

(5) This part is not given in the momentum equation, because it is negligible. However it is taken into account in the numerical program:

$$- \int_{\Omega^e} \vec{v} \cdot \rho \vec{u} \cdot (\vec{\nabla} \vec{u}) d\Omega^e \Rightarrow - \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{v}_{j,Gu}^T \rho \underline{L}_{9_j} \underset{\sim}{u}_{j,Gu}$$

$$\text{with } \underline{L}_{9_j} = \begin{bmatrix} u_1 \frac{\partial}{\partial r} + u_2 \frac{\partial}{\partial z} & 0 \\ 0 & u_1 \frac{\partial}{\partial r} + u_2 \frac{\partial}{\partial z} \end{bmatrix}_j .$$

The derivative of G towards the variable $\boldsymbol{\tau}_i$ is as follows:

$$\delta_\tau G = \int_{\Omega^e} \underbrace{- \boldsymbol{D}_v : \delta \boldsymbol{\tau}_i}_{(1)} d\Omega^e \ . \quad \text{(A.34)}$$

In a discretized matrix formulation:

(1) $- \int_{\Omega^e} \boldsymbol{D}_v : \delta \boldsymbol{\tau}_i d\Omega^e \Rightarrow - \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{v}_{j,Gu}^T \underline{B}_j^T \delta \underset{\sim}{\tau}_{i_{j,Gu}}$.

The derivative towards the variable $\vec{u}$ is as follows:

$$\delta_u G = \int_{\Omega^e} \Bigg\{ \underbrace{- \boldsymbol{D}_v : \left(2\eta_0 + 2\beta \sum_{i=1}^{N} \eta_i \right) \delta_u \boldsymbol{D}_u}_{(1)} \underbrace{- \vec{v} \cdot \rho \delta\vec{u} \cdot (\vec{\nabla}\vec{u})}_{(2)} \\ \underbrace{- \vec{v} \cdot \rho \vec{u} \cdot (\vec{\nabla}\delta\vec{u})}_{(3)} \Bigg\} d\Omega^e \ . \quad \text{(A.35)}$$

The discretized matrix formulations of the different parts are given as:

(1) $- \int_{\Omega^e} \boldsymbol{D}_v : (2\eta_0 + 2\beta \sum \eta_i)\, \delta_u \boldsymbol{D}_u d\Omega^e$

$$\Rightarrow - \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{v}_{j,Gu}^T (2\eta_0 + 2\beta \sum \eta_i)\, \underline{B}_j^T \underline{L}_{20} \underline{B}_j \delta \underset{\sim}{u}_{j,Gu},$$

(2) $- \int_{\Omega^e} \vec{v} \cdot \rho \delta\vec{u} \cdot (\vec{\nabla}\vec{u}) d\Omega^e \Rightarrow - \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{v}_{j,Gu}^T \rho \underline{L}_j \delta \underset{\sim}{u}_{j,Gu}$ with $\underline{L}_j = \begin{bmatrix} \frac{\partial u_1}{\partial r} & \frac{\partial u_1}{\partial z} \\ \frac{\partial u_2}{\partial r} & \frac{\partial u_2}{\partial z} \end{bmatrix}_j$,

(3) $- \int_{\Omega^e} \vec{v} \cdot \rho \vec{u} \cdot (\vec{\nabla}\delta\vec{u}) d\Omega^e \Rightarrow - \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{v}_{j,Gu}^T \rho \underline{L}_{9_j} \delta \underset{\sim}{u}_{j,Gu}$.

The derivative of G towards p is given by:

$$\delta_p G = \int_{\Omega^e} \underbrace{\vec{\nabla} \cdot \vec{v} \delta p}_{(1)} d\Omega^e \ . \quad \text{(A.36)}$$

In a discretized matrix formulation:

(1) $\int_{\Omega^e} \vec{\nabla} \cdot \vec{v} \delta p d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{v}_{j,Gu}^T \underline{V}_j^T \delta \underset{\sim}{p}_{j,Gu}$.

The derivative towards $\bar{\boldsymbol{D}}$ is as follows:

$$\delta_{\bar{D}} G = \int_{\Omega^e} \underbrace{\boldsymbol{D}_v : 2\beta \sum_{i=1}^{N} \eta_i \delta \bar{\boldsymbol{D}}}_{(1)} d\Omega^e \ . \quad \text{(A.37)}$$

The discretized matrix formulation is given as:

(1) $\int_{\Omega^e} \boldsymbol{D}_v : (2\beta \sum \eta_i) \delta \bar{\boldsymbol{D}} d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{v}_{j,Gu}^T (2\beta \sum \eta_i) \underline{B}_j^T \underline{L}_{30} \delta \underset{\sim}{\bar{D}}_{j,Gu}$.

The integrals must hold for all admissible test functions $\underset{\sim}{v}$. Thus, in total, the discretized matrix formulations of the equation of conservation of momentum is given by:

$$\begin{aligned}\underset{\sim}{G} = -\sum_{j=1}^{n_{intp}} w_{j,Gu}\underline{\varphi}_{j,Gu}^T \Big\{ \Big(2\eta_0 + 2\beta\sum\eta_i\Big)\,\underline{B}_j^T\underline{L}_{20}\underline{B}_j\underline{\varphi}_{j,Gu}\underset{\sim}{u}_{LGLu} \\ -(2\beta\sum\eta_i)\underline{B}_j^T\underline{L}_{30}\underline{\chi}_{j,Gu}\underset{\sim}{\bar{D}}_{LGLd} + \underline{B}_j^T\sum\underline{\mu}_{i_{j,Gu}}\underset{\sim}{\tau}_{i_{LGLt}} \\ +\rho\underline{L}_{9_j}\underline{\varphi}_{j,Gu}\underset{\sim}{u}_{LGLu} + \underline{V}_j^T\underline{\psi}_{j,Gu}\underset{\sim}{p}_{Gp}\Big\}\ , \end{aligned} \tag{A.38}$$

$$\delta_\tau\underline{G}\ \delta\underset{\sim}{\tau}_i = -\sum_{j=1}^{n_{intp}} w_{j,Gu}\underline{\varphi}_{j,Gu}^T\underline{B}_j^T\underline{\mu}_{j,Gu}\delta\underset{\sim}{\tau}_{i_{LGLt}}\ , \tag{A.39}$$

$$\begin{aligned}\delta_u\underline{G}\ \delta\underset{\sim}{u} = -\sum_{j=1}^{n_{intp}} w_{j,Gu}\underline{\varphi}_{j,Gu}^T \Big\{ \Big(2\eta_0 + 2\beta\sum\eta_i\Big)\,\underline{B}_j^T\underline{L}_{20}\underline{B}_j \\ +\rho(\underline{L}_{9_j} + \underline{L}_j)\Big\}\underline{\varphi}_{j,Gu}\delta\underset{\sim}{u}_{LGLu}\ , \end{aligned} \tag{A.40}$$

$$\delta_p\underline{G}\ \delta\underset{\sim}{p} = \sum_{j=1}^{n_{intp}} w_{j,Gu}\underline{\varphi}_{j,Gu}^T\underline{V}_j^T\underline{\psi}_{j,Gu}\delta\underset{\sim}{p}_{Gp}\ , \tag{A.41}$$

$$\delta_{\bar{D}}\underline{G}\ \delta\underset{\sim}{\bar{D}} = \sum_{j=1}^{n_{intp}} w_{j,Gu}\underline{\varphi}_{j,Gu}^T(2\beta\sum\eta_i)\underline{B}_j^T\underline{L}_{30}\underline{\chi}_{j,Gu}\delta\underset{\sim}{\bar{D}}_{LGLd}\ . \tag{A.42}$$

The inner product over an element of the extra equation for the discrete rate of deformation gives:

$$E = \int_{\Omega^e} \{\underbrace{\boldsymbol{G} : \bar{\boldsymbol{D}}}_{(1)} - \underbrace{\boldsymbol{G} : \boldsymbol{D}_u}_{(2)}\} d\Omega^e \ . \tag{A.43}$$

The discretized matrix formulations of the different parts are given as:

(1) $\int_{\Omega^e} \boldsymbol{G} : \bar{\boldsymbol{D}} d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{G}^T_{j,Gu} \underset{\sim}{\bar{D}}_{j,Gu}$,

(2) $-\int_{\Omega^e} \boldsymbol{G} : \boldsymbol{D}_u d\Omega^e \Rightarrow -\sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{G}^T_{j,Gu} \underline{L}^T_{25_j} \underset{\sim}{u}_{j,Gu}$ with $\underline{L}_{25_j} = \begin{bmatrix} \frac{\partial}{\partial r} & 0 & \frac{\partial}{\partial z} \\ 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial r} \end{bmatrix}_j$.

The derivative of E towards $\vec{u}$ is:

$$\delta_u E = -\int_{\Omega^e} \underbrace{\boldsymbol{G} : \delta_u \boldsymbol{D}_u}_{(1)} d\Omega \ . \tag{A.44}$$

The discretized matrix formulation is:

(1) $-\int_{\Omega^e} \boldsymbol{G} : \delta_u \boldsymbol{D}_u d\Omega^e \Rightarrow -\sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{G}^T_{j,Gu} \underline{L}^T_{25_j} \delta \underset{\sim}{u}^T_{j,Gu}$.

The derivative towards $\bar{\boldsymbol{D}}$ is as follows:

$$\delta_{\bar{D}} E = \int_{\Omega^e} \underbrace{\boldsymbol{G} : \delta \bar{\boldsymbol{D}}}_{(1)} d\Omega^e \ . \tag{A.45}$$

In a discretized matrix formulation:

(1) $\int_{\Omega^e} \boldsymbol{G} : \delta \bar{\boldsymbol{D}} d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{G}^T_{j,Gu} \delta \underset{\sim}{\bar{D}}_{j,Gu}$.

The other derivatives are equal to zero.

The integrals must hold for all admissible test functions $\underset{\sim}{G}$. So, in total, the discretized matrix formulations of the extra equation for the discrete rate of deformation is given by:

$$\underline{\underline{E}} = \sum_{j=1}^{n_{intp}} w_{j,Gu} \underline{\chi}^T_{j,Gu} \left\{ \underline{\chi}_{j,Gu} \underset{\sim}{\bar{D}}_{LGLd} - \underline{L}^T_{25_j} \underline{\varphi}_{j,Gu} \underset{\sim}{u}_{LGLu} \right\} \ , \tag{A.46}$$

$$\delta_\tau \underline{\underline{E}} \ \delta \underset{\sim}{\tau}_i = \underline{\underline{0}} \ , \tag{A.47}$$

$$\delta_u \underline{\underline{E}} \ \delta \underset{\sim}{u} = -\sum_{j=1}^{n_{intp}} w_{j,Gu} \underline{\chi}^T_{j,Gu} \underline{L}^T_{25_j} \underline{\varphi}_{j,Gu} \delta \underset{\sim}{u}_{LGLu} \ , \tag{A.48}$$

$$\delta_p \underline{\underline{E}} \ \delta \underline{p} = \underline{\underline{0}} \ , \tag{A.49}$$

$$\delta_{\bar{D}} \underline{\underline{E}} \ \delta \underline{\bar{D}} = \sum_{j=1}^{n_{intp}} w_{j,Gu} \underline{\chi}^T_{j,Gu} \underline{\chi}_{j,Gu} \delta \underset{\sim}{\bar{D}}_{LGLd} \ . \tag{A.50}$$

The inner product over an element of the equation of conservation of mass is given by:

$$H = \int_{\Omega^e} \underbrace{q\vec{\nabla} \cdot \vec{u}}_{(1)} d\Omega^e \ . \tag{A.51}$$

In a discretized matrix formulation:

(1) $\int_{\Omega^e} q\vec{\nabla} \cdot \vec{u} d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{q}_{j,Gu}^T \underline{V}_j \underset{\sim}{u}_{j,Gu}$.

The derivative of H towards $\vec{u}$ is given as:

$$\delta_u H = \int_{\Omega^e} \underbrace{q\vec{\nabla} \cdot \delta\vec{u}}_{(1)} d\Omega^e \ . \tag{A.52}$$

The discretized matrix formulation is given as:

(1) $\int_{\Omega^e} q\vec{\nabla} \cdot \delta\vec{u} d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{q}_{j,Gu}^T \underline{V}_j \delta\underset{\sim}{u}_{j,Gu}$.

The solution is solved with the help of a penalty function. The derivative of H towards p is therefore not equal to zero:

$$\delta_p H = \int_{\Omega^e} \underbrace{q\kappa\delta p}_{(1)} d\Omega^e \ . \tag{A.53}$$

In a discretized matrix formulation this reads:

(1) $\int_{\Omega^e} q\kappa\delta p d\Omega^e \Rightarrow \sum_{j=1}^{n_{intp}} w_{j,Gu} \underset{\sim}{q}_{j,Gu}^T \kappa\delta\underset{\sim}{p}_{j,Gu}$.

The integrals must hold for all admissible test functions $\underset{\sim}{q}$. In total, the equation of conservation of mass in a discretized matrix formulation is given as:

$$\underline{\underline{H}} = \sum_{j=1}^{n_{intp}} w_{j,Gu} \underline{\psi}_{j,Gu}^T \underline{V}_j \underline{\varphi}_{j,Gu} \underset{\sim}{u}_{LGLu} \ , \tag{A.54}$$

$$\delta_\tau \underline{\underline{H}} \ \delta\underline{\tau}_i = \underline{0} \ , \tag{A.55}$$

$$\delta_u \underline{\underline{H}} \ \delta\underline{\underline{u}} = \sum_{j=1}^{n_{intp}} w_{j,Gu} \underline{\psi}_{j,Gu}^T \underline{V}_j \underline{\varphi}_{j,Gu} \delta\underset{\sim}{u}_{LGLu} \ , \tag{A.56}$$

$$\delta_p \underline{\underline{H}} \ \delta\underline{\underline{p}} = \sum_{j=1}^{n_{intp}} w_{j,Gu} \underline{\psi}_{j,Gu}^T \kappa\underline{\psi}_{j,Gu} \delta\underset{\sim}{p}_{Gp} \ , \tag{A.57}$$

$$\delta_{\bar{D}} \underline{\underline{H}} \ \delta\underline{\bar{D}} = \underline{0} \ . \tag{A.58}$$