Pragmatism and the finite element method in the modern era

by

Rezgar Shakeri

B.S., Iran University of Science and Technology, 2011

M.S., Iran University of Science and Technology, 2013

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Civil, Environmental, and Architectural Engineering

2024

Committee Members:

Richard A. Regueiro, Chair

Jed Brown

Jeong-Hoon Song

Yida Zhang

Robert MacCurdy

Rezgar Shakeri, (Ph.D., Civil Engineering)

Pragmatism and the finite element method in the modern era

Thesis directed by Prof. Jed Brown

Since numerical modeling of hyperelastic materials is ubiquitous in many engineering fields, a main effort in computational mechanics is to develop stable, accurate and efficient tools for their analysis. A stable numerical calculation of a well-conditioned function will have a relative accuracy of $\epsilon_{\text{machine}}$. Many open source finite element analysis packages contain numerically unstable formulations at some point, even for simple material models. We trace the source of this instability and show how to formulate various compressible/incompressible hyperelastic constitutive models in a stable way.

For an incompressible material, the hydrostatic stress becomes an independent variable. Thus, it can be discretized independently of the displacement field. One of the common approaches of simulating the incompressible model is mixed finite element discretization. Mixed finite element method is an efficient approach to overcome locking that is observed in the numerical treatment of nearly incompressible materials when a pure displacement method with linear element is used. In order to guarantee the stability and optimal convergence of a mixed formulation, the inf-sup conditions must be satisfied. The inf-sup constants, which are computable through a set of eigenvalue problems for a given mixed discretization, determines the stability of the chosen FE space for displacement and pressure fields.

Despite the accuracy gained by high-order mixed finite element method, most of the industrial finite element problems are constructed within the framework of matrix-based approach by using at most quadratic/linear order of interpolation for displacement/pressure fields. A matrix-free

approach yields better performance, both in terms of storage and solve time over the assembled sparse matrices method. We introduce a matrix-free p-multigrid mixed finite element discretization with Newton-Krylov iterative solvers for the hyperelasticity problem. We implemented our problem in Ratel which is a performance portable solid mechanics library that uses matrix-free operators from libCEED and solvers from PETSc to provide fast, efficient, and accurate simulations on next-generation architectures. To demonstrate the reliability and efficiency of the mixed high-order finite element implementation, a series of numerical examples are investigated with p-multigrid preconditioning and different interpolation orders.

Dedication

To my family especially my wife.

Acknowledgements

I would like to express my sincere gratitude to my academic advisor Professor Jed Brown for his patience, kindness, and help throughout my research. He has been beyond what is required of an advisor. His help and friendship have made a huge difference in my knowledge and experience in both academic and personal life.

Contents

Chapter

Tables

Table

Figures

Figure

Chapter 1

Introduction

Numerous soft materials can be modeled by hyperelastic models, such as rubber-like elastomers, biological tissues, and bending actuators employed in soft robotics. This subject combines several challenging aspects, requiring non-linear material laws (to account for effects of large strain), geometric non-linearities (to account for large deformations), and accurate modelling of incompressibility, or slight compressibility, of the underlying materials, all for complex three-dimensional geometries. Along with understanding the characteristics of such materials, having accurate, efficient and robust numerical simulation of hyperelastic structures could also help us to predict the behavior of these materials in many engineering fields.

The standard displacement-based finite element formulations exhibits severe stiffening which leads to slow convergence, and one must use very fine meshes to get acceptable results near the incompressible limit [45]. This phenomenon is commonly referred to as volumetric locking. Several methods have been developed to improve the performance of finite elements. One of the popular pure displacement-based methods that perform well near the incompressible limit is reduced/selective integration methods or $\bar{B}$ technique [48]. In this method the volumetric term in the stiffness matrix are integrated using a lower order of Gauss integration compared with other terms. [61] showed that selective integration can be equivalent to a mixed formulation, with the number of integration points used to evaluate the volumetric contributions equal to the number of pressure degrees of freedom used in the mixed formulation. The major numerical developments on $\bar{F}$, the

generalization of $\bar{B}$ methods to the hyperelastic finite deformation formulation, have been done by [70, 33].

Another most common and efficient approach to overcome the locking is the mixed Galerkin finite element formulation. In a mixed method, besides the displacement, one or more additional fields (typically pressure) are treated as independent variables. However, the order of approximation for both pressure and displacement variables cannot be chosen arbitrarily. The mixed method must fulfill the Ladyzhenskaya–Babuška–Brezzi (LBB) or inf–sup [21, 27, 10] condition to ensure stability and optimal convergence. The mixed displacement-pressure (u/p) formulation with different interpolation orders for incompressible elastic materials can be traced back to Sussman and Bathe [87]. They demonstrate the various examples solved by u/p formulation to solve in small and finite deformation with different inf-sup stable mixed elements. Srinivasan and Perucchio [86] used the mixed model to analyze the anisotropic incompressible hyperelastic materials in 3D using linear and constant interpolations for displacement and pressure fields (Q1P0 element). They confirmed that Q1P0 element does not pass the LBB condition but they got excellent results in the their numerical experiments. However, when the instability arises in their implementations, they used quadratic and linear interpolations polynomial for displacement and pressure (Q2P1 element) which passes the LBB condition. It should be noted element that satisfies the inf-sup condition may exhibit instability in large deformation [7] but one should at least use the element that satisfies LBB condition for linear elasticity [72]. To solve a system with equal-order interpolation, the mixed model must be used in conjunction with stabilization techniques. Stabilized methods are very popular class of methods and have been utilized with success in the finite element literature for fluid mechanics and the two most commonly used are Streamline Upwind Petrov–Galerkin (SUPG) [23, 46] and Galerkin Least Squares (GLS) [47]. These ideas have been successfully extended to non-linear solid mechanics [56, 62, 38]. It is worth mentioning that another mixed method so-called enhanced strain method is also developed by [84] which adds internal element strain modes to enrich the element predictive capability. However [72] observed that this method does not perform well

in the finite deformation range and suffers from instabilities due to development of hourglass mode in some loading cases. The appearance of an hourglass mode in the context of nonlinear enhanced strain methods has first been observed by [30, 95] and numerous methods have been proposed to overcome instability [91, 89, 75]. More recently, Gurev et al. [41] simulated a high-resolution heart model with complex boundary as an incompressible hyperelastic materials with mixed pressure-displacement element. Due to limitation of the direct solver for large numbers of degrees of freedom (dof), they used a Krylov subspace iterative method along with preconditioner created based on block matrix decomposition involving the Schur complement. However, they used low-order element and the code is not optimized. Connolly et al. [28] proposed a finite element implementation in terms of principal stretches for isotropic hyperelastic rubber-like materials in initial and current configurations. They investigate their implemetation by defining the UMAT subroutines in commercial FEM software Abaqus. Furthermore, they validated their formulation with constitutive models described in terms of Cauchy-Green invariants and compared the solve times. Despite the fact that their formulation needs to solve the eigenvalue problem to find principal stretches at each Newton iteration, the computed solve time is acceptable compare with invariants-based formulation. It should be noted they used C3D8H element (Q1P0) linear and constant interpolation for displacement and pressure which is not inf-sup stable. Q1P0 element with and without the augmented Lagrangian method also examined for incompressible anisotropic hyperelastic materials by [40]. Karabelas et al [55] used perturbed Lagrangian-multiplier functional to derive the mixed formulation for an anisotropic, and nearly incompressible materials. To overcome the volumetric locking, they utilized stabilization techniques and demonstrated the high robustness and computational efficiency for two benchmark problems and a heart model as well. They discretized the problem with low-order element with a high resolution mesh which results in large block system. With suitable preconditioner and solver available in open-source software PETSc [9] the block system is solved and strong scaling was achieved for the coarse/fine grid problem up to 16/1024 cores on a desktop machine/Archer2. Apart from classical FEM, it is worth mentioning that the virtual element method (VEM) which is an extension of the classical Galerkin finite element method also

is extended for hyperelastic incompressible materials by [94, 93, 17]. VEM is not restricted by the choice of element shapes, nor is it necessary for the element to be convex. However, the construction of a virtual element which is based on projection of the deformation onto a specific ansatz space would lead to a rank deficient element once the number of vertices is greater than 3. Thus the formulation has to be stabilized. Moreover, [52] used isogeometric analysis for two-field mixed variational formulations in both small and large strain for nearly incompressible materials. Also, there has been recent interest [35, 83] in augmented Lagrangian methods with technical multigrid methods involving expensive vertex-star patch smoothers to solve incompressible problems. These methods yield low iteration counts at arbitrary element order, but are significantly more costly than standard multigrid methods and it's unclear how to achieve high performance on GPUs.

Despite the fact that all these studies with several approaches for modeling hyperelastic material in the compressible and incompressible regimes, the improvement of the performance and accuracy for is still an active area of the research. In the presented work, numerical modelling of nearly incompressible and incompressible hyperelastic is implemented in matrix-free way in Ratel which is Extensible, Performance-Portable Solid Mechanics solver. Ratel utilizes the Portable, Extensible Toolkit for Scientific Computation (PETSc), along with the libCEED library for efficient compiler optimized tensor-product-basis computation to demonstrate an efficient nonlinear solution algorithm. Our new approach shares some similarity with existing matrix-free implementation for solid mechanics [66, 26], however, the mixed matrix-free for incompressible hyperelatic material has not yet been established to the knowledge of the author. We considered three most common constitutive laws namely, Neo-Hookean, Mooney-Rivlin and Ogden for our model. In chapter 2, we briefly explain the derivation of constitutive laws and difference between these models. Then, we show the standard formulation of constitutive equations which are extensively used in many open-source and commercial finite element software packages are numerically unstable in small deformation. We trace the source of this instability and introduce a stable constitutive formulations which are stable for all range of deformation elasticity. In chapter 3, we present the

locking phenomenon arises in linear elasticity and introduce general mixed-formulation in small strain which is an alternative strategy to overcome the locking. In finite strain, first we present general mixed formulation and then we discuss a perturbed Lagrange-multiplier approach, which is based strain energy function with quadratic volumetric term. Both approaches are developed in initial and current configurations. Then, we present the notation and details of discrete matrix-free implementation for general mixed formulation. High-order matrix-free finite element operators offer superior performance on modern high performance computing hardware when compared to assembled sparse matrices [2, 37, 57], both with respect to the number of floating point operations needed for operator evaluation and the memory transfer needed for a matrix-vector product. Further, high-order finite element discretizations offer accuracy advantages over low-order finite elements [31, 71, 74]. However, high-order matrix-free operators require iterative solvers, such as Krylov subspace methods [43, 82], but these iterative solvers converge slowly for the ill-conditioned operators that come from high-order discretizations. Preconditioning techniques can significantly improve the convergence of these iterative solvers for high-order matrix-free finite element operators. p-multigrid, developed by by Rønquist and Patera [77], is particularly well suited for preconditioning problems on unstructured meshes. For mixed matrix-free formulation, we propose an upper triangular block preconditioner where we apply p-multigrid and jacobi preconditioners on the displacement and pressure blocks.

In chapter 4, we investigate the convergence order of the linear elasticity for materials with Poisson's ratios 0.375. For mixed linear case, first we compute the inf-sup constant for different mixed elements with continuous and discontinuous pressure space, then solve a MMS problem to measure the convergence order for displacement and pressure fields. Moreover, to check the performance of the proposed block preconditioner in small strain, linear iteration count, condition number and the range of eigenvalues of the preconditioned system for full Dirichlet (MMS) and clamp-traction boundary conditions are investigated. In finite strain, we perform different benchmark problems in bending and compression modes to study the performance of the mixed elements

with continuous and discontinuous pressure field in compressible and incompressible regimes and compared with displacement-based hyperelastic formulation. For all experiments, we perform convergence study and verify our results with previous studies [67, 15] and commercial finite element software ABAQUS.

Chapter 2

Stable numerics for finite-strain elasticity

A backward stable numerical calculation of a function with condition number κ will have a relative accuracy of $\kappa\epsilon_{\text{machine}}$. Standard formulations and software implementations of finite-strain elastic materials models make use of the deformation gradient $\boldsymbol{F} = I + \partial \boldsymbol{u}/\partial \boldsymbol{X}$ and Cauchy-Green tensors. These formulations are not numerically stable, leading to loss of several digits of accuracy when used in the small strain regime, and often precluding the use of single precision floating point arithmetic. We trace the source of this instability to specific points of numerical cancellation, interpretable as ill-conditioned steps. We show how to compute various strain measures in a stable way and how to transform common constitutive models to their stable representations, formulated in either initial or current configuration. The stable formulations all provide accuracy of order $\epsilon_{\text{machine}}$. In many cases, the stable formulations have elegant representations in terms of appropriate strain measures and offer geometric intuition that is lacking in their standard representation. We show that algorithmic differentiation can stably compute stresses so long as the strain energy is expressed stably, and give principles for stable computation that can be applied to inelastic materials.

2.1 Introduction

Errors in computational mechanics are attributable to three sources: continuum model specification (materials, geometry, boundary conditions), discretization (finite elements), and numerical. When working in double precision with direct solvers, the first two typically dominate numerical

errors and stable numerics are overlooked beyond linear algebra. Meanwhile, single precision is widely considered to be insufficient for finite-strain implicit analysis and practitioners opt for distinct small-strain analysis in part due to perceived cost or instability of finite-strain formulations in small-strain regimes. This shifts a cognitive burden to the practitioner who must confirm that the small-strain formulations are valid, and is problematic for high-contrast materials in which finite strains and infinitesimal strains are present within the same analysis. In this paper, we demonstrate the instability in standard formulations for hyperelasticity and present intuitive (and mathematically equivalent) reformulations that are stable, enabling finite-strain analysis at all strains and opening the door for reduced precision analysis.

In floating point arithmetic, $\epsilon_{\text{machine}} = \sup_x |\text{fl}(x) - x|/|x|$ is the maximum error incurred rounding a real number x to its nearest floating point representation $\text{fl}(x)$. (We assume x is within the exponent range and will not discuss overflow/underflow/denormals.) Typical values of $\epsilon_{\text{machine}}$ are $2^{-53} \approx 10^{-16}$ for IEEE-754 double precision and $2^{-24} \approx 6 \cdot 10^{-8}$ for single precision. Elementary math operators $\circledast$ (standing for addition, subtraction, multiplication, or division) and special functions behave as "exact arithmetic, correctly rounded", $x \circledast y = \text{fl}(x * y)$ [90], thus guaranteeing $|x \circledast y - x * y|/|x * y| \leq \epsilon_{\text{machine}}$.

With such strong guarantees from elementary operations, we might hope that $(x \circledast y) \circledast z$ is also accurate to $\epsilon_{\text{machine}}$, but also, this is not true. An illustrative example is computing $(x+1)-1$. For $x = 10^{-16}$ in double precision, $x \oplus 1 = 1$, thus $(x \oplus 1) \ominus 1 = 0$ has a relative error of 1, which is much larger than $\epsilon_{\text{machine}}$. For larger values of x, we observe the stair-step effect in Figure 2.1. The first operation incurred a relative error smaller than $\epsilon_{\text{machine}}$ and the second operation was exact, so where can we place blame for the catastrophic cancellation error? To shed light on this, we consider the condition number of a differentiable function $f(x)$, defined as

$$\kappa_f(x) = \left|\frac{\partial f}{\partial x}\right| \frac{|x|}{|f(x)|}. \tag{2.1}$$

The first operation $x + 1$ has a condition number of 1 while the second has enormous condition

number. An numerical algorithm $\tilde{f}$ for evaluating the continuum function f is called **backward stable** if it computes the exact answer to a problem with almost the given inputs, i.e., $\tilde{f}(x) = f(x')$ for some nearby x' satisfying $|x' - x|/|x| < c\epsilon_{\text{machine}}$ for a small constant c. Backward stable algorithms satisfy the forward error bound [90]

$$\frac{|\tilde{f}(x) - f(x)|}{|f(x)|} < c\kappa_f(x)\epsilon_{\text{machine}}. \tag{2.2}$$

Furthermore, this bound composes when the underlying functions are well-conditioned: if $\tilde{f}$ and $\tilde{g}$ are both backward stable algorithms, then

$$\frac{|\tilde{f}(\tilde{g}(x)) - f(g(x))|}{|f(g(x))|} < c\kappa_f(g(x))\kappa_g(x)\epsilon_{\text{machine}}.$$

Note that if $\kappa_{f\circ g}(x) \approx \kappa_f(g(x))\kappa_g(x)$, this is the bound we would get for $\tilde{f} \circ \tilde{g}$ as a backward stable algorithm. As a corollary, any calculation constructed from backward stable parts (such as elementary arithmetic) that exhibits large errors must have ill-conditioned steps. An intuitive and quantifiable strategy for designing stable algorithms is to ensure that every step is as well-conditioned as possible.

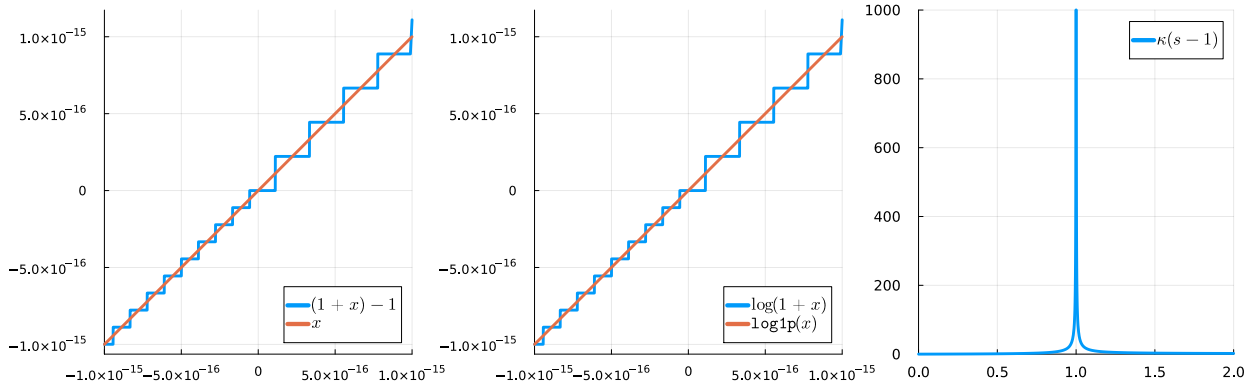


Figure 2.1: The stair-step patterns illustrate numerical instability evaluating in $(1 + x) - 1$ (left) and $\log(1 + x)$ (center). When evaluating $(1 + x) - 1$, a small relative error is incurred in the well-conditioned operation $s := 1 + x$, and that error becomes a large relative error because $s - 1$ has unbounded condition number as $s \to 1$ (right), despite this operation being computed exactly in floating point arithmetic.

We now turn our attention to representative functions in solid mechanics. Most strain-energy functions and corresponding stress models in hyperelasticity contain expressions like $\log(1 + x)$ or

$\exp(x) - 1$, which are numerically unstable expressions when $x \approx 0$. Numerical analysts proposed [12, 11]

$$\texttt{log1p}(x) = \log(1 + x) \qquad\qquad \texttt{expm1}(x) = \exp(x) - 1,$$

which gives high precision value for small values of x; these are in most core math libraries since C99. Figure 2.1 shows the $x, (1+x)-1$ and $\log(1+x), \texttt{log1p}(x)$ functions around 0. In both cases, large relative error is incurred by an algorithmic step that maps values near 1 to values near 0 via a function of derivative about 1, leading to unbounded condition number (2.1). Note that $\log(1 + x)$ and $\exp(x) - 1$ are both well conditioned, but their direct evaluation is numerically unstable due to an ill-conditioned step. We surveyed many open source finite element analysis packages and found that all contain numerically unstable formulations at some point due to phenomena explained above.

2.1.1 Finite-strain mechanics

Let $\boldsymbol{X}$ be the reference configuration and $\boldsymbol{x} = \boldsymbol{X} + \boldsymbol{u}$ be the current configuration expressed in terms of the displacement $\boldsymbol{u}$. Wriggers [92] discusses the displacement gradient

$$\boldsymbol{H} = \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{X}}$$

and mentions that the Green-Lagrange strain can be expressed as

$$\boldsymbol{E}(\boldsymbol{H}) = \frac{1}{2}(\boldsymbol{H} + \boldsymbol{H}^T + \boldsymbol{H}^T\boldsymbol{H}) \tag{2.3}$$

"in analytical investigations", but notes "This is actually not necessary when a numerical approach is applied." The usual presentation defines the deformation gradient $\boldsymbol{F} = \boldsymbol{I} + \boldsymbol{H}$ and proceeds to the right Cauchy-Green tensors $\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F}$ and $\boldsymbol{E} = \frac{1}{2}(\boldsymbol{C} - \boldsymbol{I})$. At small strain, $\boldsymbol{E}$ is small despite $\boldsymbol{F}$ and $\boldsymbol{C}$ being of order 1, leading to instability as in the examples above. For stable compressible formulations, one must also formulate expressions involving $J = |\boldsymbol{F}|$ and the first and second invariants of $\boldsymbol{C}$ in a stable way.

Developers of FEAP [88] observed numerical stability issues when applying the standard approach at small strains and have begun to favor working directly with the displacement gradient $\boldsymbol{H}$ [39] but did not complete a stable compressible hyperelastic formulation. The Abaqus UMAT [85] interface provides the deformation gradient $\boldsymbol{F}$ and a strain increment, but does not provide direct access to the displacement gradient $\boldsymbol{H}$ or nonlinear strain tensor, therefore numerical cancellation is inevitable when small strains appear within large strain formulations. FEBio [60], a nonlinear finite element package for biomechanical applications, uses the deformation gradient $\boldsymbol{F}$ even for defining the linear strain tensor. Moreover, they offer a varieties of hyperelasticity models that contains the subtraction, which leads to loss of significance when the problem is in a small deformation regime. MoFEM [51], an open source library for solving complex physics problems, uses the standard formulation given in continuum mechanics text books to describe the stress-strain relation of material. This constitutive formulation contains $\log J$ function, leading to instability when $J \approx 1$. Similarly, the Multiphysics Object-Oriented Simulation Environment (MOOSE) [58] defines, for example, the Neo-Hookean model using the standard (numerically unstable) formulation. Albany-LCM [79], a finite element code for analysis of multiphysics problem on unstructured grid expresses the hyperelastic models in terms of $J^2 - 1$, which leads to catastrophic cancellation when $J \approx 1$. Table 2.1 summarizes the stability properties of formulations used in well-known text books and production software.

The paper proceeds as follows: section 2.2 develops stable formulations for common hyperelastic models in coupled and decoupled (isochoric and volumetric split), section 2.3 demonstrates that stable stress expressions can be derived using algorithmic differentiation (AD) so long as care is taken in the strain energy formulation, and section 2.4 concludes with outlook toward inelastic models. Details of the numerical procedure for evaluating stability are given in Appendix B. All figures exhibited here are created using the open source Julia programming language. Comprehensive numerical experiments and figures are provided in the executable supplement [81] for those readers who are interested in exploring of all given formulations here. While this study presents

Table 2.1: A stable constitutive implementation requires stable computation of strain $\boldsymbol{E}$, $J = |\boldsymbol{F}|$, $\log J$, and shear expressions involving first and second invariants of $\boldsymbol{C}$. Unstable formulations are widespread in books and software, with Ratel having the only stable formulation we are aware of.

Software/book	stable strain $\boldsymbol{E}$	stable J	stable $\log J$	stable constitutive equation
FEAP [88]	✓	–	–	–
FEBio [60]	–	–	–	–
Abaqus [85]	–	–	–	–
MOOSE [58]	–	–	–	–
Albany-LCM [79]	–	–	–	–
LifeV [14]	–	–	–	–
MoFEM [51]	–	–	–	–
Ratel [6]	✓	✓	✓	✓
Holzapfel [44]	–	–	–	–
Wriggers [92]	–	–	–	–

some of the most well-known hyperelastic models and their stable formulation, the approach is general and we provide guidance for applying it to other material models.

2.2 Constitutive equations

The constitutive behavior for hyperelastic materials is characterized by a strain energy density function ψ. For isotropic materials in initial configuration, ψ is typically defined by either the principal invariants $\{\mathbb{I}_1, \mathbb{I}_2, \mathbb{I}_3\}$ of right Cauchy-Green tensor $\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F}$ or the principal stretches $\{\lambda_1, \lambda_2, \lambda_3\}$ of the SPD matrix $\boldsymbol{U}$ where $\boldsymbol{RU} = \boldsymbol{F}$ is the polar decomposition. In the following we discuss the most common coupled and decoupled representation of the strain energies and the associated constitutive equations that are employed frequently in the literature.

2.2.1 Coupled strain energy

Coupled strain energy functionals ψ are written in terms of invariants without an isochoric-volumetric split. In the linear regime, this corresponds to use of shear modulus μ and first Lamé parameter λ, with the standard (not deviatoric) infinitesimal strain tensor $\boldsymbol{\varepsilon}$. For the general form of coupled strain energy

$$\psi\left(\boldsymbol{C}\right) = \psi\left(\mathbb{I}_1, \mathbb{I}_2, \mathbb{I}_3\right), \tag{2.4}$$

with the invariants

$$\begin{aligned}\mathbb{I}_1(\boldsymbol{C}) &= \operatorname{trace}\boldsymbol{C},\\ \mathbb{I}_2(\boldsymbol{C}) &= \frac{1}{2}\left(\mathbb{I}_1^2 - \boldsymbol{C}:\boldsymbol{C}\right)\\ \mathbb{I}_3(\boldsymbol{C}) &= |\boldsymbol{C}|.\end{aligned} \tag{2.5}$$

we can determine the constitutive equations by taking the gradient of strain energy as

$$\boldsymbol{S} = \frac{\partial\psi}{\partial\boldsymbol{E}} = 2\frac{\partial\psi}{\partial\boldsymbol{C}} = 2\sum_{i=1}^{3}\frac{\partial\psi}{\partial\mathbb{I}_i}\frac{\partial\mathbb{I}_i}{\partial\boldsymbol{C}}, \tag{2.6}$$

where $\boldsymbol{S}$ is the general form of the stress relation in initial configuration and

$$\begin{aligned}\frac{\partial\mathbb{I}_1}{\partial\boldsymbol{C}} &= \boldsymbol{I},\\ \frac{\partial\mathbb{I}_2}{\partial\boldsymbol{C}} &= \mathbb{I}_1\boldsymbol{I} - \boldsymbol{C}\\ \frac{\partial\mathbb{I}_3}{\partial\boldsymbol{C}} &= \mathbb{I}\boldsymbol{C}^{-1}.\end{aligned} \tag{2.7}$$

In the following we introduce the stable formulation for two well-known hyperelastic constitutive equations.

2.2.1.1 Neo-Hookean model

One of the simplest hyperelastic models is the Neo-Hookean model, given by

$$\psi(\boldsymbol{C}) = \frac{\lambda}{4}\left(J^2 - 1 - 2\log J\right) - \mu\log J + \frac{\mu}{2}\left(\mathbb{I}_1 - 3\right) \tag{2.8}$$

where $J = |\boldsymbol{F}| = \sqrt{|\boldsymbol{C}|}$. The first term is a convex choice satisfying limit conditions [92, 32] while the second is a structural necessity for coupled strain energy formulations. The second Piola-Kirchhoff stress is derived according to (2.7)

$$\boldsymbol{S} = 2\frac{\partial\psi}{\partial\boldsymbol{C}} = \frac{\partial\psi}{\partial\boldsymbol{E}} = \frac{\lambda}{2}\left(J^2 - 1\right)\boldsymbol{C}^{-1} + \mu\left(\boldsymbol{I} - \boldsymbol{C}^{-1}\right), \tag{2.9}$$

where $\boldsymbol{C} = \boldsymbol{I} + 2\boldsymbol{E}$. Both terms are numerically unstable expressions at small strain ($\boldsymbol{C} \approx I$), but the first is also unstable when $J \approx 1$ at large strain, a condition that is prevalent for nearly incompressible materials. Similar to `log1p`, we need a formulation that avoids direct computation of J in $J^2 - 1 = (J-1)(J+1)$.

Consider the 2-dimensional case

$$\boldsymbol{F} = \boldsymbol{I} + \boldsymbol{H} = \boldsymbol{I} + \nabla_X \boldsymbol{u} = \begin{bmatrix} 1 + u_{1,1} & u_{1,2} \\ u_{2,1} & 1 + u_{2,2} \end{bmatrix},$$

and let $\mathtt{J}_{-1} = J - 1$ be computed by the stable expression

$$\mathtt{J}_{-1} = u_{1,1} + u_{2,2} + u_{1,1}u_{2,2} - u_{1,2}u_{2,1}. \tag{2.10}$$

Remark 1. *Ideally, one would like a proof of backward stability for every constitutive model, but doing so is tedious and we believe offers no great insight. In light of* (2.2)*, it is revealing to plot the forward relative error and observe that stable algorithms provide errors that are uniformly of order* $\epsilon_{machine}$ *(sometimes written "in* $O(\epsilon_{machine})$*") independent of the input argument* $\boldsymbol{H}$ *(or a strain). There is a caveat: in the case of tiny perturbations of a pure rotation* $\boldsymbol{H} \approx \boldsymbol{Q} - \boldsymbol{I}$ *for an orthogonal matrix* $\boldsymbol{Q}$*, the strain is nearly zero and thus stress will also be nearly zero. For example, the Green-Lagrange strain* $\boldsymbol{E}(\boldsymbol{H}) = \frac{1}{2}(\boldsymbol{F}^T\boldsymbol{F} - \boldsymbol{I}) \approx \frac{1}{2}(\boldsymbol{Q}^T\boldsymbol{Q} - \boldsymbol{I}) = \boldsymbol{0}$ *is a function with unbounded condition number, thus even a backward stable algorithm such as* (2.3) *will have unbounded error in* $\boldsymbol{E}$*. One might consider high/mixed precision or alternative state variables* $\boldsymbol{u}$ *that make strain a well-conditioned function of the state if a problem requires small-strain stability for large motions of floating bodies. While we consider* $\boldsymbol{H}$ *as the input in our numerical experiments (to be agnostic over strain measures), we will not construct cases of nearly-pure rotations.*

Remark 2. *Apart from the incompressible limit, hyperelastic constitutive models are wellconditioned, thus* (2.2) *ensures the error will be in* $O(\epsilon_{machine})$*. Although various displacement-only formulations are common in engineering practice, mixed methods are necessary for well-conditioned finite element formulations in the incompressible limit.*

Using a 3-dimensional analog of (2.10) described in Appendix B and numerical procedure explained therein, Figure 2.2 shows that the relative error in $\mathtt{J}_{-1}$ is $O(\epsilon_{\text{machine}})$ independent of the magnitude of the displacement gradient $\boldsymbol{H}$ and strain, while $J - 1$ loses digits of accuracy at small strain. Replacing $J^2 - 1$ with $\mathtt{J}_{-1}(\mathtt{J}_{-1} + 2)$ is sufficient to stabilize the first term of (2.9).

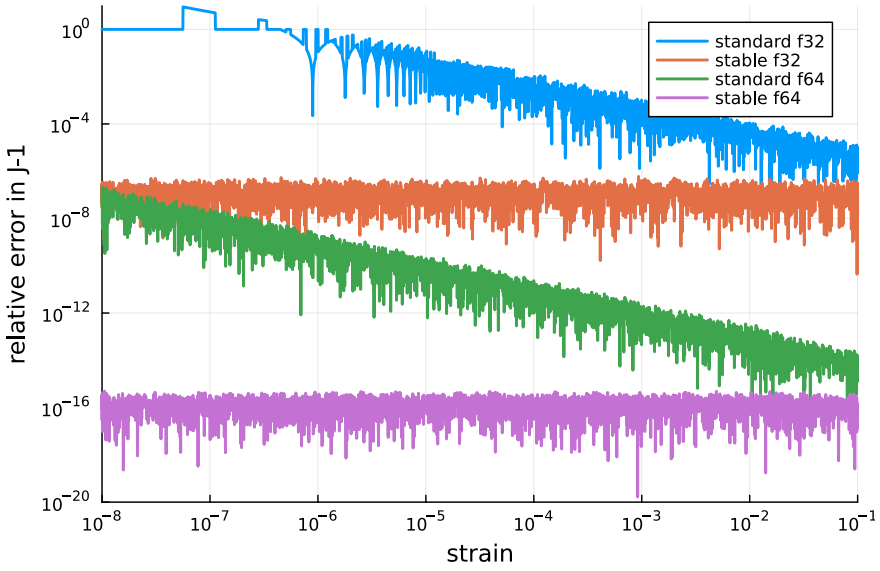


Figure 2.2: Relative error of standard computation of $J - 1$ and its stable way $\mathtt{J}_{-1}$

In some constitutive models, $\log J$ appears directly in the expression for $\boldsymbol{S}$, and also has a huge condition number $\kappa_{\log}(J) = 1/\log J$ when $J \approx 1$. In such cases, one naturally achieves stability via

$$\log J = \texttt{log1p}\left(\texttt{J}_{-1}\right). \tag{2.11}$$

The Neo-Hookean constitutive equation (2.9) is also unstable due to cancellation in the second term when $\boldsymbol{C}^{-1} \approx \boldsymbol{I}$, and it can be fixed by replacing $\boldsymbol{I} - \boldsymbol{C}^{-1} = \boldsymbol{C}^{-1}(\boldsymbol{C} - \boldsymbol{I}) = 2\boldsymbol{C}^{-1}\boldsymbol{E}$. Thus, an equivalent stable form of (2.9) is

$$\boldsymbol{S} = \frac{\lambda}{2}\texttt{J}_{-1}\left(\texttt{J}_{-1} + 2\right)\boldsymbol{C}^{-1} + 2\mu\boldsymbol{C}^{-1}\boldsymbol{E}. \tag{2.12}$$

The relative error in $\boldsymbol{S}$ using the standard (2.9) and stable (2.12) expressions yields a figure indistinguishable from Figure 2.2. Although we elide figures for each constitutive model and stress, the supplement [81] contains numerical demonstration that every expression we call unstable or stable exhibits errors indistinguishable from the corresponding line in Figure 2.2.

The Neo-Hookean constitutive model can also be evaluated in current configuration, classically written in terms of J and the left Cauchy-Green tensor $\boldsymbol{b} = \boldsymbol{F}\boldsymbol{F}^T$. One can derive an expression for a current configuration stress (we use Kirchhoff stress $\boldsymbol{\tau}$) starting with a strain energy density $\psi(\boldsymbol{b})$ or (equivalently) pushing $\boldsymbol{S}$ of (2.9) forward via

$$\boldsymbol{\tau} = \boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^T = \frac{\lambda}{2}\left(J^2 - 1\right)\boldsymbol{I} + \mu\left(\boldsymbol{b} - \boldsymbol{I}\right). \tag{2.13}$$

A stable form analogous to (2.12) is

$$\boldsymbol{\tau} = \frac{\lambda}{2}\texttt{J}_{-1}\left(\texttt{J}_{-1} + 2\right)\boldsymbol{I} + 2\mu\boldsymbol{e}, \tag{2.14}$$

where we call $\boldsymbol{e} = \frac{1}{2}(\boldsymbol{b} - \boldsymbol{I}) = \frac{1}{2}(\boldsymbol{H} + \boldsymbol{H}^T + \boldsymbol{H}\boldsymbol{H}^T)$ the Green-Euler strain tensor. (It is the $n = 2$ current configuration Seth-Hill strain measure [44].)

2.2.1.2 Mooney-Rivlin model

Compared to the Neo-Hookean model (2.8), the coupled energy function for Mooney-Rivlin model depends on the additional invariant $\mathbb{I}_2(\boldsymbol{C}) = \frac{1}{2}\left(\mathbb{I}_1^2(\boldsymbol{C}) - \boldsymbol{C}:\boldsymbol{C}\right)$ where $\boldsymbol{C}:\boldsymbol{C} = \text{trace}(\boldsymbol{C}^T\boldsymbol{C})$ is the Frobenius inner product. The coupled Mooney-Rivlin strain energy is given by [44]

$$\psi(J,\boldsymbol{C}) = \frac{\lambda}{4}\left(J^2 - 1 - 2\log J\right) - (\mu_1 + 2\mu_2)\log J + \frac{\mu_1}{2}\left(\mathbb{I}_1(\boldsymbol{C}) - 3\right) + \frac{\mu_2}{2}\left(\mathbb{I}_2(\boldsymbol{C}) - 3\right), \tag{2.15}$$

where μ_1 and μ_2 are parameters that must be experimentally determined and $\mu = \mu_1 + \mu_2$ is the shear modulus in the linear regime. The second Piola-Kirchhoff tensor is

$$\boldsymbol{S} = \frac{\lambda}{2}\left(J^2 - 1\right)\boldsymbol{C}^{-1} + \mu_1\left(\boldsymbol{I} - \boldsymbol{C}^{-1}\right) + \mu_2\left(\mathbb{I}_1(\boldsymbol{C})\boldsymbol{I} - 2\boldsymbol{C}^{-1} - \boldsymbol{C}\right). \tag{2.16}$$

Similar to the Neo-Hookean model, we write the stable form in terms of Green-Lagrange strain,

$$\boldsymbol{S} = \frac{\lambda}{2}\mathtt{J}_{-1}\left(\mathtt{J}_{-1} + 2\right)\boldsymbol{C}^{-1} + 2\left(\mu_1 + 2\mu_2\right)\boldsymbol{C}^{-1}\boldsymbol{E} + 2\mu_2\left(\mathbb{I}_1(\boldsymbol{E})\boldsymbol{I} - \boldsymbol{E}\right). \tag{2.17}$$

In addition, the Kirchhoff stress tensor for Mooney-Rivlin in current configuration is classically written

$$\boldsymbol{\tau} = \frac{\lambda}{2}\left(J^2 - 1\right)\boldsymbol{I} + \mu_1\left(\boldsymbol{b} - \boldsymbol{I}\right) + \mu_2\left(\mathbb{I}_1(\boldsymbol{b})\boldsymbol{b} - 2\boldsymbol{I} - \boldsymbol{b}^2\right) \tag{2.18}$$

and its stable form in terms of Green-Euler strain is

$$\boldsymbol{\tau} = \frac{\lambda}{2}\mathtt{J}_{-1}\left(\mathtt{J}_{-1} + 2\right)\boldsymbol{I} + 2\left(\mu_1 + 2\mu_2\right)\boldsymbol{e} + 2\mu_2\left(\mathbb{I}_1(\boldsymbol{e})\boldsymbol{I} - \boldsymbol{e}\right)\boldsymbol{b}. \tag{2.19}$$

2.2.2 Decoupled strain energy

Decoupled strain energy formulations decompose the strain into volumetric and deviatoric parts. In the linear regime, such formulations use the shear modulus μ and bulk modulus k, with the deviatoric strain $\boldsymbol{\varepsilon}_{\text{dev}} = \boldsymbol{\varepsilon} - \frac{1}{3}(\text{trace}\,\boldsymbol{\varepsilon})\boldsymbol{I}$. In the case of rubber-like materials, the bulk modulus k is orders of magnitude larger than the shear modulus, $k \gg \mu$. For finite strain, the strain energy function is split into volumetric and isochoric (volume-preserving) parts. The first step is multiplicative decomposition of the deformation gradient as

$$\boldsymbol{F} = (J^{1/3}\boldsymbol{I})\bar{\boldsymbol{F}} = J^{1/3}\bar{\boldsymbol{F}} \tag{2.20}$$

where $J^{1/3}\boldsymbol{I}$ describes the purely volumetric deformation and $\bar{\boldsymbol{F}}$ captures the isochoric since $|\bar{\boldsymbol{F}}| = |J^{-1/3}\boldsymbol{F}| = 1$. Similarly we can decompose right Cauchy-Green tensor

$$\bar{\boldsymbol{C}} = \bar{\boldsymbol{F}}^T\bar{\boldsymbol{F}} = J^{-2/3}\boldsymbol{C} \tag{2.21}$$

where $\bar{\boldsymbol{F}}$ and $\bar{\boldsymbol{C}}$ are known as the modified deformation gradient and modified right Cauchy-Green tensor, respectively. Moreover, the modified principal stretches $\bar{\lambda}_i$ and modified invariants $\bar{\mathbb{I}}_i$ can be defined

$$\bar{\lambda}_i = J^{-1/3}\lambda_i,\ i = 1, 2, 3 \tag{2.22}$$

$$\bar{\mathbb{I}}_1 = J^{-2/3}\mathbb{I}_1, \qquad \bar{\mathbb{I}}_2 = J^{-4/3}\mathbb{I}_2, \qquad \bar{\mathbb{I}}_3 = \left(J^{-2/3}\right)^3\mathbb{I}_3 = 1. \tag{2.23}$$

The general decoupled strain energy is a sum of volumetric and isochoric parts

$$\psi\left(J, \bar{\boldsymbol{C}}\right) = \psi_{\text{vol}}\left(J\right) + \psi_{\text{iso}}\left(\bar{\boldsymbol{C}}\right), \tag{2.24}$$

leading to an additive decomposition of the second Piola-Kirchhoff stress

$$\boldsymbol{S} = \frac{\partial\psi}{\partial\boldsymbol{E}} = \frac{\partial\psi_{\text{vol}}}{\partial J}\frac{\partial J}{\partial\boldsymbol{E}} + \frac{\partial\psi_{\text{iso}}}{\partial\boldsymbol{E}} = \boldsymbol{S}_{\text{vol}} + \boldsymbol{S}_{\text{iso}}. \tag{2.25}$$

When the isochoric strain energy is written in terms of modified invariants $\psi_{\text{iso}}(\bar{\mathbb{I}}_1, \bar{\mathbb{I}}_2)$ (note that $\bar{\mathbb{I}}_3 = 1$ uniformly), the isochoric stress satisfies

$$\boldsymbol{S}_{\text{iso}} = \frac{\partial\psi_{\text{iso}}}{\partial\boldsymbol{E}} = 2\frac{\partial\psi_{\text{iso}}}{\partial\boldsymbol{C}} = 2\sum_{i=1}^{3}\frac{\partial\psi_{\text{iso}}}{\partial\bar{\mathbb{I}}_i}\frac{\partial\bar{\mathbb{I}}_i}{\partial\boldsymbol{C}} \tag{2.26}$$

with

$$\begin{aligned}\frac{\partial\bar{\mathbb{I}}_1}{\partial\boldsymbol{C}} &= \frac{\partial(J^{-2/3}\mathbb{I}_1)}{\partial\boldsymbol{C}} = J^{-2/3}\left(\boldsymbol{I} - \frac{1}{3}\mathbb{I}_1\boldsymbol{C}^{-1}\right)\\ \frac{\partial\bar{\mathbb{I}}_2}{\partial\boldsymbol{C}} &= \frac{\partial(J^{-4/3}\mathbb{I}_2)}{\partial\boldsymbol{C}} = J^{-4/3}\left(\mathbb{I}_1\boldsymbol{I} - \boldsymbol{C} - \frac{2}{3}\mathbb{I}_2\boldsymbol{C}^{-1}\right),\end{aligned} \tag{2.27}$$

where we have used $\frac{\partial J}{\partial\boldsymbol{C}} = \frac{1}{2}J\boldsymbol{C}^{-1}$. As discussed for (2.9), there are many empirical forms for the pure volumetric part (or bulk term) of the strain-energy, which should be convex and satisfy physical constrains [32, 68]. We consider one such form,

$$\psi_{\text{vol}} = \frac{k}{4}\left(J^2 - 1 - 2\log J\right), \tag{2.28}$$

but similar principles will give stable formulations for others. The volumetric stress can be defined as

$$\boldsymbol{S}_{\text{vol}} = -p_{\text{hyd}} \frac{\partial J}{\partial \boldsymbol{E}} = -p_{\text{hyd}} J \boldsymbol{C}^{-1} \tag{2.29}$$

where we have used the definition of hydrostatic pressure $p_{\text{hyd}} = -\frac{\partial \psi_{\text{vol}}}{\partial J}$, which for (2.28) can be stably computed via

$$p_{\text{hyd}} = -\frac{k}{2J}\left(J^2 - 1\right) = -\frac{k}{2J}\mathtt{J}_{-1}\left(\mathtt{J}_{-1} + 2\right). \tag{2.30}$$

While p_{hyd} depends on the form $\psi_{\text{vol}}(J)$, the volumetric stress (2.29) is numerically stable and always the same expression in terms of p. In the decoupled framework, the only salient difference between the Neo-Hookean, Mooney-Rivlin, and Ogden models is in their isochoric part, thus we focus on stable expressions for isochoric stresses $\boldsymbol{S}_{\text{iso}}$ and $\boldsymbol{\tau}_{\text{iso}}$.

2.2.2.1 Neo-Hookean model

The decoupled strain energy density for the Neo-Hookean model is

$$\psi(J, \bar{\boldsymbol{C}}) = \psi_{\text{vol}}(J) + \frac{\mu}{2}\left(\bar{\mathbb{I}}_1 - 3\right) \tag{2.31}$$

The isochoric second Piola-Kirchhoff stress is a straightforward application of (2.27),

$$\boldsymbol{S}_{\text{iso}} = \mu J^{-2/3}\left(\boldsymbol{I} - \frac{1}{3}\mathbb{I}_1 \boldsymbol{C}^{-1}\right). \tag{2.32}$$

Using the relation $\mathbb{I}_1(\boldsymbol{C}) = 3 + 2\mathbb{I}_1(\boldsymbol{E})$, the numerically stable form of (2.32) can be written as

$$\boldsymbol{S}_{\text{iso}} = 2\mu J^{-2/3}\boldsymbol{C}^{-1}\left(\boldsymbol{E} - \frac{1}{3}\mathbb{I}_1(\boldsymbol{E})\boldsymbol{I}\right) = 2\mu J^{-2/3}\boldsymbol{C}^{-1}\boldsymbol{E}_{\text{dev}}, \tag{2.33}$$

which makes use of the deviatoric Green-Lagrange strain $\boldsymbol{E}_{\text{dev}} = \boldsymbol{E} - \frac{1}{3}\mathbb{I}_1(\boldsymbol{E})\boldsymbol{I}$.

In current configuration, the isochoric Kirchhoff stress is

$$\boldsymbol{\tau}_{\text{iso}} = \boldsymbol{F}\boldsymbol{S}_{\text{iso}}\boldsymbol{F}^T = \mu J^{-2/3}\left(\boldsymbol{b} - \frac{1}{3}\mathbb{I}_1(\boldsymbol{b})\boldsymbol{I}\right) \tag{2.34}$$

and its equivalent stable form is

$$\boldsymbol{\tau}_{\text{iso}} = 2\mu J^{-2/3}\left(\boldsymbol{e} - \frac{1}{3}\mathbb{I}_1(\boldsymbol{e})\boldsymbol{I}\right) = 2\mu J^{-2/3}\boldsymbol{e}_{\text{dev}}, \tag{2.35}$$

where we have used $\mathbb{I}_1(\boldsymbol{b}) = 3 + 2\mathbb{I}_1(\boldsymbol{e})$.

2.2.2.2 Mooney-Rivlin model

For the Mooney-Rivlin model, decoupled strain energy density is given by

$$\psi(J,\bar{\boldsymbol{C}}) = \psi_{\text{vol}}(J) + \frac{\mu_1}{2}\left(\bar{\mathbb{I}}_1 - 3\right) + \frac{\mu_2}{2}\left(\bar{\mathbb{I}}_2 - 3\right) \tag{2.36}$$

Using (2.27), the isochoric second Piola-Kirchhoff stress can be written as

$$\boldsymbol{S}_{\text{iso}} = \mu_1 J^{-2/3}\left(\boldsymbol{I} - \frac{1}{3}\mathbb{I}_1(\boldsymbol{C})\boldsymbol{C}^{-1}\right) + \mu_2 J^{-4/3}\left(\mathbb{I}_1(\boldsymbol{C})\boldsymbol{I} - \boldsymbol{C} - \frac{2}{3}\mathbb{I}_2(\boldsymbol{C})\boldsymbol{C}^{-1}\right) \tag{2.37}$$

and its stable form is

$$\begin{aligned}
\boldsymbol{S}_{\text{iso}} &= 2\mu_1 J^{-2/3}\boldsymbol{C}^{-1}\left(\boldsymbol{E} - \frac{1}{3}\mathbb{I}_1(\boldsymbol{E})\boldsymbol{I}\right) + 2\mu_2 J^{-4/3}\left(\mathbb{I}_1(\boldsymbol{E})\boldsymbol{I} - \boldsymbol{E}\right) \\
&\quad + 4\mu_2 J^{-4/3}\boldsymbol{C}^{-1}\left(\boldsymbol{E} - \frac{2}{3}\mathbb{I}_1(\boldsymbol{E})\boldsymbol{I} - \frac{2}{3}\mathbb{I}_2(\boldsymbol{E})\boldsymbol{I}\right) \\
&= 2\left(\mu_1 J^{-2/3} + 2\mu_2 J^{-4/3}\right)\boldsymbol{C}^{-1}\boldsymbol{E}_{\text{dev}} + 2\mu_2 J^{-4/3}\left(\mathbb{I}_1(\boldsymbol{E})\boldsymbol{I} - \boldsymbol{E}\right) \\
&\quad - \frac{4}{3}\mu_2 J^{-4/3}\left(\mathbb{I}_1(\boldsymbol{E}) + 2\mathbb{I}_2(\boldsymbol{E})\right)\boldsymbol{C}^{-1},
\end{aligned} \tag{2.38}$$

where we have used $\mathbb{I}_1(\boldsymbol{C}) = 3 + 2\mathbb{I}_1(\boldsymbol{E})$, $\mathbb{I}_2(\boldsymbol{C}) = 3 + 4\mathbb{I}_1(\boldsymbol{E}) + 4\mathbb{I}_2(\boldsymbol{E})$.

In current configuration we have

$$\boldsymbol{\tau}_{\text{iso}} = \mu_1 J^{-2/3}\left(\boldsymbol{b} - \frac{1}{3}\mathbb{I}_1(\boldsymbol{b})\boldsymbol{I}\right) + \mu_2 J^{-4/3}\left(\mathbb{I}_1(\boldsymbol{b})\boldsymbol{b} - \boldsymbol{b}^2 - \frac{2}{3}\mathbb{I}_2(\boldsymbol{b})\boldsymbol{I}\right) \tag{2.39}$$

and the stable form of (2.39) is

$$\begin{aligned}
\boldsymbol{\tau}_{\text{iso}} &= 2\mu_1 J^{-2/3}\left(\boldsymbol{e} - \frac{1}{3}\mathbb{I}_1(\boldsymbol{e})\boldsymbol{I}\right) + 2\mu_2 J^{-4/3}\left(\mathbb{I}_1(\boldsymbol{e})\boldsymbol{I} - \boldsymbol{e}\right)\boldsymbol{b} + 4\mu_2 J^{-4/3}\left(\boldsymbol{e} - \frac{2}{3}\mathbb{I}_1(\boldsymbol{e})\boldsymbol{I} - \frac{2}{3}\mathbb{I}_2(\boldsymbol{e})\boldsymbol{I}\right) \\
&= 2\left(\mu_1 J^{-2/3} + 2\mu_2 J^{-4/3}\right)\boldsymbol{e}_{\text{dev}} + 2\mu_2 J^{-4/3}\left(\mathbb{I}_1(\boldsymbol{e})\boldsymbol{I} - \boldsymbol{e}\right)\boldsymbol{b} - \frac{4}{3}\mu_2 J^{-4/3}\left(\mathbb{I}_1(\boldsymbol{e}) + 2\mathbb{I}_2(\boldsymbol{e})\right)\boldsymbol{I}
\end{aligned} \tag{2.40}$$

where we have used $\mathbb{I}_1(\boldsymbol{b}) = 3 + 2\mathbb{I}_1(\boldsymbol{e})$, $\mathbb{I}_2(\boldsymbol{b}) = 3 + 4\mathbb{I}_1(\boldsymbol{e}) + 4\mathbb{I}_2(\boldsymbol{e})$.

2.2.2.3 Ogden model

The postulated decoupled strain energy for Ogden model is a function of the modified principal stretches (2.22) as [92, 44]

$$\psi(J,\bar{\lambda}_i) = \psi_{\text{vol}}(J) + \psi_{\text{iso}}(\bar{\lambda}_i) = \psi_{\text{vol}}(J) + \sum_{i=1}^{3} \bar{\omega}(\bar{\lambda}_i) \quad \text{with} \quad \bar{\omega}(\bar{\lambda}_i) = \sum_{j=1}^{N} \frac{\mu_j}{\alpha_j}\left(\bar{\lambda}_i^{\alpha_j} - 1\right) \tag{2.41}$$

where the parameters μ_j and α_j have to be determined from experiments. In the linearized regime, all hyperelastic models reduce to linear elasticity, where the shear modulus μ satisfies

$$2\mu = \sum_{j=1}^{N} \mu_j \alpha_j \quad \text{with} \quad \mu_j \alpha_j > 0 \tag{2.42}$$

in terms of the Ogden parameters. The isochoric second Piola-Kirchhoff stress can be written as

$$\boldsymbol{S}_{\text{iso}} = 2\frac{\partial \psi_{\text{iso}}(\bar{\lambda}_i)}{\partial \boldsymbol{C}} = \sum_{i=1}^{3} 2\frac{\partial \psi_{\text{iso}}}{\partial \lambda_i^2}\frac{\partial \lambda_i^2}{\partial \boldsymbol{C}} = \sum_{i=1}^{3} \frac{1}{\lambda_i}\frac{\partial \psi_{\text{iso}}}{\partial \lambda_i}\boldsymbol{N}_i\boldsymbol{N}_i^T = \sum_{i=1}^{3} s_i^{\text{iso}}\boldsymbol{N}_i\boldsymbol{N}_i^T \tag{2.43}$$

where an eigenvector $\boldsymbol{N}_i$ is computed by $\boldsymbol{C}\boldsymbol{N}_i = \lambda_i^2\boldsymbol{N}_i$. By employing $\frac{\partial J}{\partial \lambda_i} = J\lambda_i^{-1}$ we have

$$s_i^{\text{iso}} = \frac{1}{\lambda_i}\frac{\partial \psi_{\text{iso}}}{\partial \lambda_i} = \frac{1}{\lambda_i}\frac{\partial \bar{\lambda}_k}{\partial \lambda_i}\frac{\partial \psi_{\text{iso}}}{\partial \bar{\lambda}_k} = \frac{J^{-1/3}}{\lambda_i}\left(\delta_{ik} - \frac{1}{3}\lambda_i^{-1}\lambda_k\right)\frac{\partial \psi_{\text{iso}}}{\partial \bar{\lambda}_k} = \frac{J^{-1/3}}{\lambda_i}\left(\delta_{ik} - \frac{1}{3}\bar{\lambda}_i^{-1}\bar{\lambda}_k\right)\frac{\partial \psi_{\text{iso}}}{\partial \bar{\lambda}_k} \tag{2.44}$$

in which

$$\frac{\partial \psi_{\text{iso}}}{\partial \bar{\lambda}_k} = \frac{\partial \bar{\omega}(\bar{\lambda}_k)}{\partial \bar{\lambda}_k} = \sum_{j=1}^{N} \mu_j \bar{\lambda}_k^{\alpha_j - 1} \tag{2.45}$$

To derive an equivalent numerically stable form of (2.43) we rewrite the ψ_{iso} by substituting $\bar{\lambda}_i = J^{-1/3}\lambda_i$ as

$$\psi_{\text{iso}}(\lambda_i) = \sum_{j=1}^{N} \frac{\mu_j}{\alpha_j}\left[\left(\lambda_1^{\alpha_j} + \lambda_2^{\alpha_j} + \lambda_3^{\alpha_j}\right) J^{-\alpha_j/3} - 3\right] \tag{2.46}$$

and computing its derivative as

$$
\begin{aligned}
s_1^{\text{iso}} = \frac{1}{\lambda_1}\frac{\partial \psi_{\text{iso}}}{\partial \lambda_1} &= \frac{1}{\lambda_1}\sum_{j=1}^{N}\frac{\mu_j}{\alpha_j}\left[\left(\alpha_j \lambda_1^{\alpha_j - 1}\right) J^{-\alpha_j/3} - \frac{\alpha_j}{3} J^{-\alpha_j/3}\lambda_1^{-1}\left(\lambda_1^{\alpha_j} + \lambda_2^{\alpha_j} + \lambda_3^{\alpha_j}\right)\right] \\
&= \frac{1}{\lambda_1}\sum_{j=1}^{N}\mu_j\left[\lambda_1^{\alpha_j - 1} - \frac{1}{3}\lambda_1^{-1}\left(\lambda_1^{\alpha_j} + \lambda_2^{\alpha_j} + \lambda_3^{\alpha_j}\right)\right] J^{-\alpha_j/3} \\
&= \frac{1}{\lambda_1^2}\sum_{j=1}^{N}\frac{\mu_j}{3}\left[2\lambda_1^{\alpha_j} - \lambda_2^{\alpha_j} - \lambda_3^{\alpha_j}\right] J^{-\alpha_j/3} \\
&= \frac{1}{\lambda_1^2}\sum_{j=1}^{N}\frac{\mu_j}{3}\left[2\left(\lambda_1^{\alpha_j} - 1\right) - \left(\lambda_2^{\alpha_j} - 1\right) - \left(\lambda_3^{\alpha_j} - 1\right)\right] J^{-\alpha_j/3} \\
&= \frac{1}{\lambda_1^2}\sum_{j=1}^{N}\frac{\mu_j}{3}\left[2(e^{\alpha_j \ell_1} - 1) - (e^{\alpha_j \ell_2} - 1) - (e^{\alpha_j \ell_3} - 1)\right] J^{-\alpha_j/3} \\
&= \frac{1}{1 + 2\lambda_1^E}\sum_{j=1}^{N}\frac{\mu_j}{3}\left[2\,\texttt{expm1}(\alpha_j \ell_1) - \texttt{expm1}(\alpha_j \ell_2) - \texttt{expm1}(\alpha_j \ell_3)\right] J^{-\alpha_j/3}
\end{aligned}
\tag{2.47}
$$

where λ_i^E is the eigenvalue of strain tensor $\boldsymbol{E}$ and $\ell_i = \log \lambda_i = \frac{1}{2}\,\texttt{log1p}(2\lambda_i^E)$. Following the above approach we have

$$
\begin{aligned}
s_2^{\text{iso}} = \frac{1}{\lambda_2}\frac{\partial \psi_{\text{iso}}}{\partial \lambda_2} &= \frac{1}{1 + 2\lambda_2^E}\sum_{j=1}^{N}\frac{\mu_j}{3}\left[-\,\texttt{expm1}(\alpha_j \ell_1) + 2\,\texttt{expm1}(\alpha_j \ell_2) - \texttt{expm1}(\alpha_j \ell_3)\right] J^{-\alpha_j/3} \\
s_3^{\text{iso}} = \frac{1}{\lambda_3}\frac{\partial \psi_{\text{iso}}}{\partial \lambda_3} &= \frac{1}{1 + 2\lambda_3^E}\sum_{j=1}^{N}\frac{\mu_j}{3}\left[-\,\texttt{expm1}(\alpha_j \ell_1) - \texttt{expm1}(\alpha_j \ell_2) + 2\,\texttt{expm1}(\alpha_j \ell_3)\right] J^{-\alpha_j/3}.
\end{aligned}
\tag{2.48}
$$

Substituting the new definition of s_i^{iso} (2.47) and (2.48) into (2.43) provides a stable form of the isochoric second Piola-Kirchhoff stress for the Ogden model. Relative error for the standard (2.44) and stable (2.47)-(2.48) form of the isochoric second Piola-Kirchhoff stress (2.43) for the Ogden model is plotted in Figure 2.3. The behavior for single and double precision is similar to Figure 2.2.

Note that for symmetric, real-valued 3×3 strain tensor $\boldsymbol{E}$, the standard closed-form solutions for eigenvalues are susceptible to loss of significance in floating point computation. Therefore we computed λ_i^E, by the stable algorithm proposed in [42], which has a relative accuracy of $\epsilon_{\text{machine}}$.

The isochoric Kirchhoff stress tensor for Ogden model is given by

$$\boldsymbol{\tau}_{\text{iso}} = \boldsymbol{F}\boldsymbol{S}_{\text{iso}}\boldsymbol{F}^T = \sum_{i=1}^{3} \tau_i^{\text{iso}} \boldsymbol{n}_i \boldsymbol{n}_i{}^T. \tag{2.49}$$

where we have used $\boldsymbol{F}\boldsymbol{N}_i = \lambda_i \boldsymbol{n}_i$ which $\boldsymbol{n}_i$ are eigenvectors of $\boldsymbol{b}$ and $\tau_i^{\text{iso}} = \lambda_i^2 s_i^{\text{iso}}$.

Similar to the initial configuration, we can compute the $\boldsymbol{\tau}_{\text{iso}}$ in stable form by using the stable coefficients

$$\begin{aligned}
\tau_1^{\text{iso}} &= \lambda_1 \frac{\partial \psi_{\text{iso}}}{\partial \lambda_1} = \sum_{j=1}^{N} \frac{m_j}{3} \left[2\,\texttt{expm1}(\alpha_j \ell_1) - \texttt{expm1}(\alpha_j \ell_2) - \texttt{expm1}(\alpha_j \ell_3)\right] J^{-\alpha_j/3} \\
\tau_2^{\text{iso}} &= \lambda_2 \frac{\partial \psi_{\text{iso}}}{\partial \lambda_2} = \sum_{j=1}^{N} \frac{m_j}{3} \left[-\texttt{expm1}(\alpha_j \ell_1) + 2\,\texttt{expm1}(\alpha_j \ell_2) - \texttt{expm1}(\alpha_j \ell_3)\right] J^{-\alpha_j/3} \\
\tau_3^{\text{iso}} &= \lambda_3 \frac{\partial \psi_{\text{iso}}}{\partial \lambda_3} = \sum_{j=1}^{N} \frac{m_j}{3} \left[-\texttt{expm1}(\alpha_j \ell_1) - \texttt{expm1}(\alpha_j \ell_2) + 2\,\texttt{expm1}(\alpha_j \ell_3)\right] J^{-\alpha_j/3}
\end{aligned} \tag{2.50}$$

and computing the eigen pair $(\lambda_i, \boldsymbol{n}_i)$ of Green-Euler strain tensor $\boldsymbol{e}\boldsymbol{n}_i = \lambda_i^e \boldsymbol{n}_i$ with $\lambda_i = \sqrt{1 + 2\lambda_i^e}$ and $\ell_i = \log \lambda_i = \frac{1}{2}\,\texttt{log1p}(2\lambda_i^e)$.

2.2.3 Hencky strain

Standard material logarithmic or Hencky strain is defined as

$$\boldsymbol{E}_{\text{H}} = \log(\boldsymbol{U}) = \frac{1}{2}\log(\boldsymbol{C}) = \frac{1}{2}\sum_{i=1}^{3} \log(\lambda_i) \boldsymbol{N}_i \boldsymbol{N}_i^T, \tag{2.51}$$

which is numerically unstable when $\lambda_i \approx 1$. The stable version can be computed by

$$\boldsymbol{E}_{\text{H}} = \frac{1}{2}\log(2\boldsymbol{E} + I) = \frac{1}{2}\sum_{i=1}^{3} \texttt{log1p}(2\lambda_i^E) \boldsymbol{N}_i \boldsymbol{N}_i^T. \tag{2.52}$$

We can use a similar approach to define the stable form of the spatial logarithmic in current configuration

$$\boldsymbol{e}_{\text{H}} = \log(\boldsymbol{v}) = \frac{1}{2}\log(\boldsymbol{b}) \tag{2.53}$$

by using the $\boldsymbol{e}$'s eigenvalues as

$$\boldsymbol{e}_{\text{H}} = \frac{1}{2}\log(2\boldsymbol{e} + I) = \frac{1}{2}\sum_{i=1}^{3} \texttt{log1p}(2\lambda_i^e) \boldsymbol{n}_i \boldsymbol{n}_i^T. \tag{2.54}$$

With strain computed in a stable way, constitutive models based on the Hencky strain can stably evaluated using the principles in the prior sections.

2.3 Algorithmic Differentiation to Derive Material Models

Deriving some of the material models in section 2.2 is not a trivial task, and could be tedious and error-prone. One way to derive complicated material models without needing to manipulate and simplify the intermediate expressions is to use algorithmic (aka. automatic) differentiation (AD). Starting from the strain energy function in terms of the strain tensor, one could use an AD tool to compute the corresponding stress tensor. However, using AD does not **automatically** guarantee stability. Instabilities in the strain energy function will propagate to the derivatives, leading to an unstable evaluation of stress. Our primary goal here is to show how to compute material models using AD, identify instability-inducing terms in the free energy function, and introducing a stable form of the strain energy function. We chose the coupled Neo-Hookean model (2.8) to show the procedure. One can apply these principles to obtain stable representations for other material models.

We start with re-writing equation (2.8) in terms of $\boldsymbol{E}$.

$$\psi(\boldsymbol{E}) = \frac{\lambda}{4}\left(J^2 - 1 - 2\log J\right) - \mu\left(\log J - \operatorname{trace}(\boldsymbol{E})\right), \tag{2.55}$$

where $J = \sqrt{|2\boldsymbol{E} + I|}$. As we saw in subsubsection 2.2.1.1, equation (2.55) is unstable due to the presence of the $J^2 - 1$ and $\log J$ terms. Using (2.10)-(2.11), we can transform (2.55) to

$$\begin{aligned}\psi(\boldsymbol{E}) &= \frac{\lambda}{4}\left(\mathtt{J}_{-1}\left(\mathtt{J}_{-1}+2\right) - 2\,\mathtt{log1p}\left(\mathtt{J}_{-1}\right)\right) - \mu\left(\mathtt{log1p}\left(\mathtt{J}_{-1}\right) - \operatorname{trace}(\boldsymbol{E})\right)\\ &= \frac{\lambda}{4}\Big(\underbrace{\mathtt{J}_{-1}{}^2}_{O(\epsilon^2)} - 2\underbrace{\left(\mathtt{log1p}\,\mathtt{J}_{-1} - \mathtt{J}_{-1}\right)}_{O(\epsilon^2)}\Big) - \mu\underbrace{\left(\mathtt{log1p}\,\mathtt{J}_{-1} - \operatorname{trace}\boldsymbol{E}\right)}_{O(\epsilon^2)}.\end{aligned} \tag{2.56}$$

While (2.56) is more stable than (2.55), it is still unstable when $J \approx 1$ and its derivative results in an unstable formulation for stress due to numerical cancellations in the $\mathtt{log1p}\,\mathtt{J}_{-1} - \mathtt{J}_{-1}$ and $\mathtt{log1p}\left(\mathtt{J}_{-1}\right) - \operatorname{trace}(\boldsymbol{E})$ terms. Looking more closely at these terms, if $\boldsymbol{E}$ is of scale ϵ, then

$\psi(\boldsymbol{E}) \in O(\epsilon^2)$ and thus we need to avoid subtracting terms such as $\mathtt{J}_{-1}$ and $\operatorname{trace}\boldsymbol{E}$, which are $O(\epsilon)$. The first underbrace in (2.56) is fine as is, but the second and third require a reformulation.

For the second underbrace, we define a helper function for computing $\log(1+x) - x$ that avoids subtracting $O(x)$ terms when computing the $O(x^2)$ result. Knowing [12]

$$\begin{aligned} \mathtt{log1p}(x) &= \log(1+x) \\ &= 2\operatorname{artanh}\left(\frac{x}{2+x}\right) \\ &= 2\sum_{n=0}^{\infty} \frac{1}{2n+1}\left(\frac{x}{2+x}\right)^{2n+1}, \quad \left|\frac{x}{2+x}\right| < 1\,. \end{aligned} \tag{2.57}$$

and moving x to the left hand side, we have

$$\begin{aligned} \mathtt{log1p\_minus\_x}(x) &= \log(1+x) - x \\ &= -\frac{x^2}{2+x} + 2\sum_{n=1}^{\infty} \frac{1}{2n+1}\left(\frac{x}{2+x}\right)^{2n+1}, \quad \left|\frac{x}{2+x}\right| < 1\,. \end{aligned} \tag{2.58}$$

section B.1 studies how many terms are necessary to evaluate `log1p_minus_x` accurately.

With a stable implementation of `log1p_minus_x` available, we need only a stable formulation for the third underbrace in (2.56). Considering a 2×2 strain tensor, we can break down J^2 into

$$\begin{aligned} J^2 &= |\boldsymbol{I} + 2\boldsymbol{E}| \\ &= 1 + 2\underbrace{(E_{1,1} + E_{2,2})}_{\operatorname{trace}(\boldsymbol{E})} + 4\left(E_{1,1}E_{2,2} - E_{1,2}E_{2,1}\right)\,, \end{aligned} \tag{2.59}$$

and define a new variable:

$$\begin{aligned} \mathtt{j} &= J^2 - 1 - 2\operatorname{trace}(\boldsymbol{E}) \\ &= 4\left(E_{1,1}E_{2,2} - E_{1,2}E_{2,1}\right)\,. \end{aligned} \tag{2.60}$$

Using (2.60) extended for 3×3 matrices, we can compute $\mathtt{J}_{-1}$ as

$$\mathtt{J}_{-1} = \frac{J^2 - 1}{J + 1} = \frac{\mathtt{j} + 2\operatorname{trace}(\boldsymbol{E})}{J+1}\,. \tag{2.61}$$

Finally, with (2.58)-(2.61), we arrive at a stable representation of the strain energy function (2.55) as

$$\psi(\boldsymbol{E}) = \frac{\lambda}{4}\left({\mathtt{J}_{-1}}^2 - 2\,\mathtt{log1p\_minus\_x}(\mathtt{J}_{-1})\right) - \mu\left(\mathtt{log1p\_minus\_x}\left(\mathtt{j} + 2\operatorname{trace}\boldsymbol{E}\right) + \mathtt{j}\right)\,. \tag{2.62}$$

Figure 2.4 shows the relative error of strain energy function for Neo-Hookean model. For the strain of order 10^{-8} we lose 16 and 8 digits in the standard (2.55) and semistable (2.56) forms, respectively, while the stable form (2.62) delivers full accuracy.

We can now expect AD tools to automatically generate a stable representation of the second Piola-Kirchhoff stress, $\boldsymbol{S} = \frac{\partial \psi}{\partial \boldsymbol{E}}$, and indeed, Figure 2.5 shows that direct application of Zygote.jl [49] to (2.62) (with $n = 6$ in (2.58)) is stable. Meanwhile, the standard and semistable forms both lose 8 digits for strain at order 10^{-8}. Listing 2.1 shows sample Julia code to compute $\boldsymbol{S}$ via reverse-mode AD. We note that AD tools that can compute higher derivatives, e.g., by applying forward-mode AD to $\boldsymbol{S}(\boldsymbol{E})$, can readily provide ingredients for solvers (such as Newton linearization) and diagnostics, freeing the implementer from tedious coding of higher derivatives or resorting to numerical differentiation.

Listing 2.1: Code for computing stress § for an arbitrary strain energy `psi`($\boldsymbol{E}$) using the AD tool Zygote.jl.

```
using Zygote
function S(E, psi)
    stress, = Zygote.gradient(psi, E)
    stress
end
```

2.4 Conclusions

In this paper, we investigated various constitutive formulations for elasticity along with their implementation in finite element software packages. Formulations written in terms of the deformation gradient $\boldsymbol{F}$ cannot be numerically stable, Standard formulations have additional instabilities due to the presence of function like $J - 1$ and/or $\log J$, which are unstable when $J \approx 1$, as well as terms like $\boldsymbol{I} - \boldsymbol{C}^{-1}$, which similarly experience cancellation for small strain $\boldsymbol{E} \approx 0$. In general, the standard computation for a strain of order 10^{-m} will result in m digits lost in the computed

stress tensor and $2m$ lost evaluating the strain energy function. We proposed equivalent stable formulations, all of which achieve relative accuracy $O(\epsilon_{\text{machine}})$. These new formulations make use of the displacement gradient $\boldsymbol{H}$ to define a strain tensor without loss of significance, compute $J-1$ in a stable way, and avoid cancelation computing shear stress terms. In addition to coupled and decoupled forms of Neo-Hookean, Mooney-Rivlin, and Ogden models in initial and current configuration, we showed that one can achieve a stable formulation using algorithmic differentiation (AD) if the forward model (strain energy) is stable. We also showed a stable evaluation of Hencky strain, which is important for developing stable representations for inelastic models at finite strain.

With single precision in the standard formulation, the first digit of stress is incorrect for strains of order 10^{-7}, while the new stable formulations get all 7 digits correct at all strain levels. The stable formulations open the door for hyperelastic simulation using single or mixed precision [1], thereby improving performance and reducing hardware and energy cost without compromising accuracy. Moreover, stable formulations are necessary to run efficiently on hardware that does not support double precision, such as GPUs, tensor cores, and embedded devices.

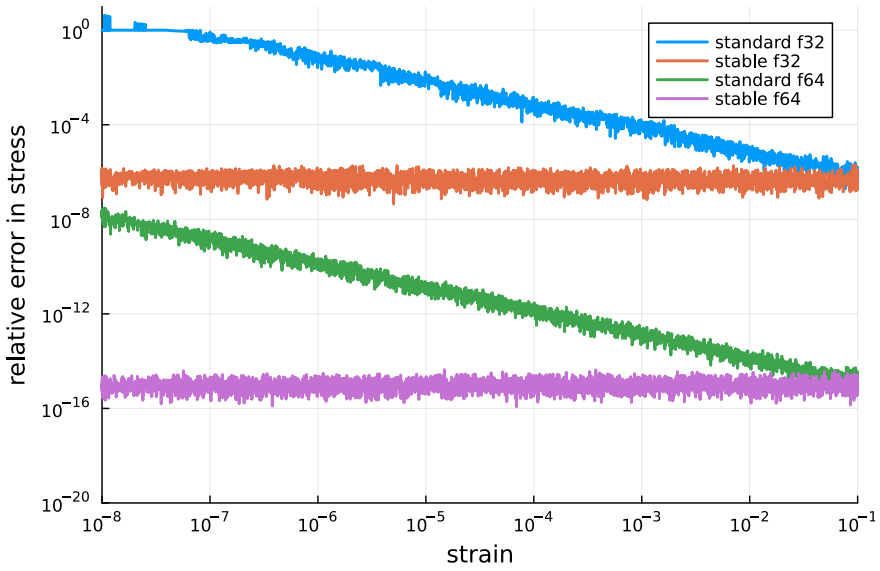


Figure 2.3: Relative error of standard computation of isochoric second Piola-Kirchhoff stress for Ogden model and its stable form

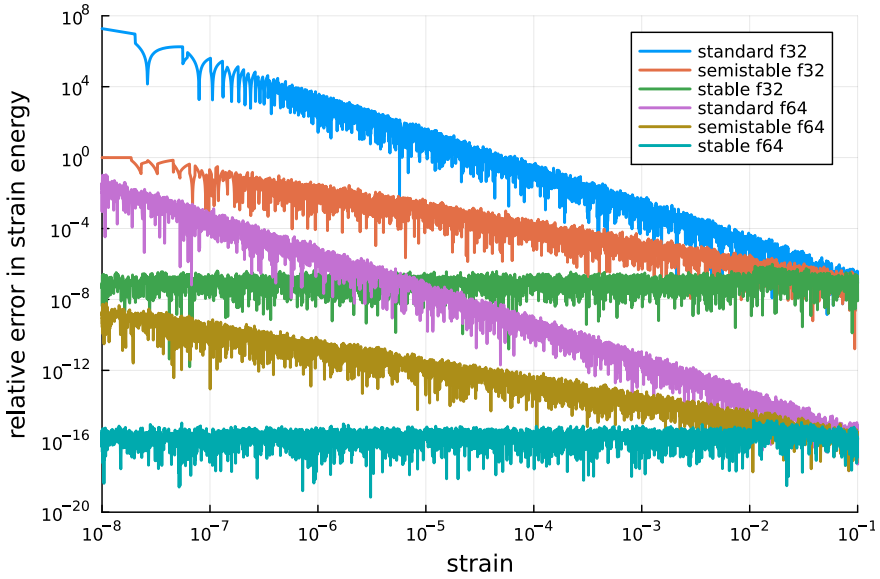


Figure 2.4: Relative error of computation of Neo-Hookean strain energy in standard (2.55), semistable (2.56) and stable (2.62) forms.

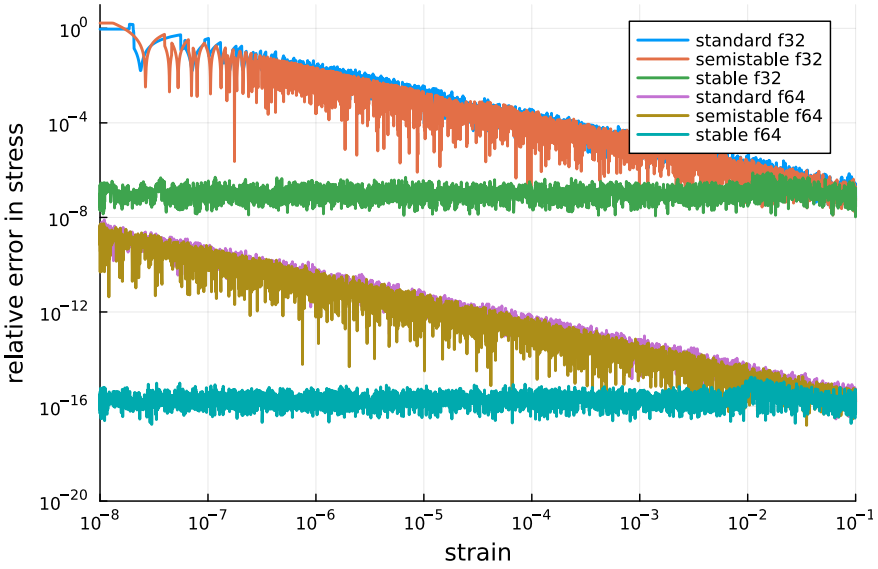


Figure 2.5: Relative error of computation of stress with AD using the standard, semistable and stable forms of strain energy.

Chapter 3

Governing Equation and Implementation

Many problems of physical importance involve motions that essentially preserve volumes locally. That is, after deformation each small portion of the medium has the same volume as before deformation. Media that behave in this fashion are termed **incompressible**. Rubber, biological tissue and many fluid flows are assumed incompressible. Single field FEM discretization leads to locking when low-order element is used. To understand the problem of locking, we will examine first how the problem arises in linear elasticity from both a theoretical and heuristic viewpoint, and then discuss how locking has been overcome in that field.

3.1 Mixed Linear Elasticity

Recall for the linear elasticity, the stress-strain relation is given by [45]

$$\boldsymbol{\sigma} = \lambda \operatorname{trace} \boldsymbol{\varepsilon}\, \boldsymbol{I} + 2\mu\boldsymbol{\varepsilon}, \tag{3.1}$$

where μ is shear modulus, λ is first Lame parameter defined in terms of Poisson's ratio ν and μ by

$$\lambda = \frac{2\nu\mu}{1-2\nu} \tag{3.2}$$

and $\boldsymbol{\varepsilon}$ is linear strain tensor

$$\boldsymbol{\varepsilon}(\boldsymbol{u}) = \frac{1}{2}\left(\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^T\right). \tag{3.3}$$

The Galerkin approximation form for linear elasticity is: Given finite dimension subspace

$\mathbb{V}_h \subset H^1(\Omega)$, find $\boldsymbol{u} \in \mathbb{V}_h$, such that for all $\boldsymbol{v} \in \mathbb{V}_h^0$

$$\int_\Omega 2\mu\, \boldsymbol{\varepsilon}(\boldsymbol{v}) : \boldsymbol{\varepsilon}(\boldsymbol{u}_h)\, dv + \int_\Omega \lambda\, \nabla \cdot \boldsymbol{v}\, \nabla \cdot \boldsymbol{u}_h\, dv = \int_\Omega \boldsymbol{v} \cdot \rho \boldsymbol{g}\, dv + \int_{\partial\Omega^N} \boldsymbol{v} \cdot \bar{\boldsymbol{t}}\, ds \tag{3.4}$$

where $\mathbb{V}_h^0 = \{\boldsymbol{v} \in \mathbb{V}_h : \boldsymbol{v} = 0 \text{ on } \partial\Omega_D\}$. The classical error bounds for the continuum solution $\boldsymbol{u}$ and the discrete solution of the displacement $\boldsymbol{u}_h$ in a sufficient regular domain is [4, 73]

$$\|\boldsymbol{u} - \boldsymbol{u}_h\| \leq C(\mu, k)\, h^{n+1}, \tag{3.5}$$

where

$$k = \frac{2\mu(1+\nu)}{3(1-2\nu)} = \lambda + \frac{2\mu}{3}, \tag{3.6}$$

is the bulk modulus, h is the mesh size, the norm on the left is usual L_2 norm, and n is polynomial order of basis function. Using higher-order polynomial and more refined meshes should reduce the discretization error. However, the constant C depends on material parameters and can become quite large and makes the error estimate practically worthless. For $\frac{\lambda}{\mu} = \mathcal{O}(10)$ (3.4) gives satisfactory convergent finite element approximations to the elasticity problem. But the performance of the finite element methods deteriorates as $\lambda \to \infty$ (or equivalently $\nu \to 0.5$). In this case, the term involving λ in (3.1), and (3.4) becomes very large and leads to non-physical results which is known as the phenomenon of *locking*. Moreover, the regularity estimate given in [19]

$$\|\boldsymbol{u}\|_{H^2} + \lambda \|\nabla \cdot \boldsymbol{u}\|_{H^1} \leq C\left(\|\rho \boldsymbol{g}\| + \|\boldsymbol{g}^{\boldsymbol{u}}\|_{H^2} + \|\bar{\boldsymbol{t}}\|_{H^1}\right) \tag{3.7}$$

where C is independent of λ and μ, also gives us a hint of trouble that leads to locking when λ is large. One can see from the above equations that as $\lambda \longrightarrow \infty$, it is necessary that $\nabla \cdot \boldsymbol{u} \longrightarrow 0$ which gives an idea of alternative strategies.

One approach is to define an auxiliary variable p [45], and rewrite constitutive equation (3.1) as

$$\begin{aligned} \boldsymbol{\sigma}(\boldsymbol{u}, p) &= -p\, \boldsymbol{I} + 2\mu \boldsymbol{\varepsilon}, \\ p &= -\lambda\, \mathrm{trace}\, \boldsymbol{\varepsilon}. \end{aligned} \tag{3.8}$$

Alternatively, we can use the definition of hydrostatic pressure [10] i.e., $p_{\text{hyd}} = -\frac{\text{trace}\,\boldsymbol{\sigma}}{3}$ and arrive at

$$\begin{aligned} \boldsymbol{\sigma}(\boldsymbol{u}, p_{\text{hyd}}) &= -p_{\text{hyd}}\,\boldsymbol{I} + 2\mu\boldsymbol{\varepsilon}_{\text{dev}}, \\ p_{\text{hyd}} &= -k\,\text{trace}\,\boldsymbol{\varepsilon}. \end{aligned} \tag{3.9}$$

where $\boldsymbol{\varepsilon}_{\text{dev}} = \boldsymbol{\varepsilon} - \frac{1}{3}\,\text{trace}\,\boldsymbol{\varepsilon}\ \boldsymbol{I}$ is the deviatoric part of the linear strain tensor and k is the bulk modulus. We present a general constitutive equation as

$$\begin{aligned} \boldsymbol{\sigma}(\boldsymbol{u}, p) &= (k_p\,\text{trace}\,\boldsymbol{\varepsilon} - p)\,\boldsymbol{I} + 2\mu\boldsymbol{\varepsilon}_{\text{dev}}, \\ p &= -\,(k - k_p)\,\text{trace}\,\boldsymbol{\varepsilon}. \end{aligned} \tag{3.10}$$

where $k_p = 2\mu\,(1+\nu_p)\,/3\,(1-2\nu_p)$ is the primal portion of the bulk modulus, defined in terms of ν_p with $-1 \leq \nu_p < \nu$, where ν is the physical Poisson's ratio. The standard full-train formulation (3.8) is obtained using $\nu_p = 0$, and the deviatoric formulation (3.9) with $\nu_p = -1$. The general mixed boundary-value problem (Strong form) for (3.10) may then be stated as follows: Given body force $\rho\boldsymbol{g}$, Dirichlet boundary $\boldsymbol{g}^{\boldsymbol{u}}$ and applied traction $\bar{\boldsymbol{t}}$, find the displacement and pressure-like variable $(\boldsymbol{u}, p) \in \mathbb{V} \times \mathbb{Q}$ (here $\mathbb{V} = H^1(\Omega), \mathbb{Q} = L^2(\Omega)$), such that:

$$\begin{aligned} -\nabla\cdot\boldsymbol{\sigma} - \rho\boldsymbol{g} &= 0, \qquad \text{in } \Omega \\ -\nabla\cdot\boldsymbol{u} - \frac{p}{k - k_p} &= 0, \qquad \text{in } \Omega \\ \boldsymbol{u} &= \boldsymbol{g}^{\boldsymbol{u}}, \quad \text{on } \partial\Omega^D \\ \boldsymbol{\sigma}\cdot\boldsymbol{n} &= \bar{\boldsymbol{t}}. \qquad \text{on } \partial\Omega^N \end{aligned} \tag{3.11}$$

with $\boldsymbol{n}$ be the unit normal on the boundary and its weak formulation as:

$$\begin{aligned} a(\boldsymbol{v}, \boldsymbol{u}) + b(\boldsymbol{v}, p) &= (\boldsymbol{v}, \rho\boldsymbol{g}) + \langle \boldsymbol{v}, \bar{\boldsymbol{t}} \rangle_{\partial\Omega^N}, \quad \forall \boldsymbol{v} \in \mathbb{V}^0 \\ b(q, \boldsymbol{u}) + c(q, p) &= 0, \quad \forall q \in \mathbb{Q} \end{aligned} \tag{3.12}$$

where

$$\begin{aligned} a(\boldsymbol{v},\boldsymbol{u}) &= \int_\Omega \nabla\boldsymbol{v} : (2\mu\,\varepsilon_{\text{dev}} + k_p \,\text{trace}\,\boldsymbol{\varepsilon})\; dv \\ b(\boldsymbol{v},p) &= -\int_\Omega \nabla\cdot\boldsymbol{v}\, p\, dv \\ c(q,p) &= -\int_\Omega q\frac{p}{k-k_p}\, dv \\ (\boldsymbol{v},\rho\boldsymbol{g}) &= \int_\Omega \boldsymbol{v}\cdot\rho\boldsymbol{g}\, dv \\ \langle\boldsymbol{v},\bar{\boldsymbol{t}}\rangle_{\partial\Omega^N} &= \int_{\partial\Omega^N} \boldsymbol{v}\cdot\bar{\boldsymbol{t}}\, ds \end{aligned} \tag{3.13}$$

The discrete version of (3.12) can be obtained by replacing $\mathbb{V},\mathbb{Q}$ by $\mathbb{V}_h \subset \mathbb{V}_h, \mathbb{Q}_h \subset \mathbb{Q}$ and $(\boldsymbol{u},p)$ by $(\boldsymbol{u}_h,p_h)$. For any discretization of a variational problem, stability is crucial to ensure well-posedness (solutions exist, are unique and depend continuously on the given data). For coercive problems, the discrete stability may often be easily ensured. For mixed discretizations of saddle point problems on the other hand, stability may be a nontrivial affair. Indeed, the mixed finite element spaces must usually be carefully chosen. The stability theory for mixed finite element discretizations originates from the work of [8] and [20] in the early 1970's. In order to have uniform behavior in the limit as $h \to 0$, the Babuska constant γ_h must be $\gamma_h \geq \gamma_0 > 0$ for all $h > 0$; that is, that γ_h is bounded from below independently of h [8]. For our mixed problem (3.12) the Babuska constant γ_h is defined by

$$0 < \gamma_0 \leq \gamma_h = \inf_{0\neq(\boldsymbol{u}_h,p_h)} \sup_{0\neq(\boldsymbol{v},q)} \frac{|d((\boldsymbol{u}_h,p_h),(\boldsymbol{v},q))|}{(\|\boldsymbol{u}_h\|_{\mathbb{V}_h} + \|p_h\|_{\mathbb{Q}_h})(\|\boldsymbol{v}\|_{\mathbb{V}} + \|q\|_{\mathbb{Q}})}. \tag{3.14}$$

where

$$d((\boldsymbol{u}_h,p_h),(\boldsymbol{v},q)) = a(\boldsymbol{v},\boldsymbol{u}_h) + b(\boldsymbol{v},p_h) + b(q,\boldsymbol{u}_h) + c(q,p_h) \tag{3.15}$$

However, the single Babuska stability condition (3.14) can be split into a pair of stability conditions as follows [20]. We define α_h as the Brezzi coercivity constant and β_h as the Brezzi

inf-sup constant by

$$\begin{aligned}\alpha_h &= \inf_{0\neq \boldsymbol{u}_h\in\mathbb{Z}_h}\sup_{0\neq \boldsymbol{v}\in\mathbb{Z}_h}\frac{|a(\boldsymbol{v},\boldsymbol{u}_h)|}{\|\boldsymbol{u}_h\|_{\mathbb{V}_h}\|\boldsymbol{v}\|_{\mathbb{V}_h}},\\ \beta_h &= \inf_{0\neq q\in\mathbb{Q}_h}\sup_{0\neq \boldsymbol{u}_h\in\mathbb{V}_h}\frac{|b(q,\boldsymbol{u}_h)|}{\|\boldsymbol{u}_h\|_{\mathbb{V}_h}\|q\|_{\mathbb{Q}_h}},\end{aligned} \tag{3.16}$$

where

$$\mathbb{Z}_h = \{\boldsymbol{u}_h\in\mathbb{V}_h | b(q,\boldsymbol{u}_h)=0 \quad \forall q\in\mathbb{Q}_h\} \tag{3.17}$$

The Brezzi stability conditions state that there exists a unique solution of (3.12) if and only if α_h and β_h (or equivalently γ_h) are bounded below for all $h>0$ [5]. Moreover, it can be shown that the Brezzi inf-sup constant is positive if and only if there are no nontrivial spurious modes [59]. These conditions enable a concise characterization of the stability of the discretized saddle point problems. Unlike (3.5) where we had explicit bounds in terms of mesh size and material properties, the error estimate for the mixed formulation [21]

$$\begin{aligned}\mu\|\boldsymbol{u}-\boldsymbol{u}_h\|_{\mathbb{V}} &\leq C\left[\frac{\mu}{\beta}\inf_{\boldsymbol{v}\in\mathbb{V}}\|\boldsymbol{u}-\boldsymbol{v}\|_{\mathbb{V}}+\inf_{q\in\mathbb{Q}}\|p-q\|_{\mathbb{Q}}\right]\\ \|p-p_h\|_{\mathbb{Q}} &\leq \frac{C}{\beta}\left[\frac{\mu}{\beta}\inf_{\boldsymbol{v}\in\mathbb{V}}\|\boldsymbol{u}-\boldsymbol{v}\|_{\mathbb{V}}+\inf_{q\in\mathbb{Q}}\|p-q\|_{\mathbb{Q}}\right]\end{aligned} \tag{3.18}$$

depends only on shear modulus μ and inf-sup constant β.

For a given discretization, the Brezzi constants are computable through a set of eigenvalue problems [5, 59]: Find $\lambda\in\mathbb{R}$, $0\neq(\boldsymbol{u}_h,p_h)\in\mathbb{V}_h\times\mathbb{Q}_h$, such that for all $(\boldsymbol{v},q)\in\mathbb{V}_h^0\times\mathbb{Q}_h$

$$a(\boldsymbol{v},\boldsymbol{u}_h)+b(\boldsymbol{v},p_h)+b(q,\boldsymbol{u}_h) = -\lambda\langle q,p_h\rangle_{\mathbb{Q}_h} \tag{3.19}$$

then, $\lambda\geq 0$ and inf-sup constant defined in (3.16) is $\beta_h=\sqrt{\lambda_{\min}}$. The coercivity constant is equal to magnitude of the smallest eigenvalue i.e., $\alpha_h=|\lambda_{\min}|$ of the following eigenvalue problem

$$a(\boldsymbol{v},\boldsymbol{u}_h)+b(\boldsymbol{v},p_h)+b(q,\boldsymbol{u}_h) = \lambda\langle\boldsymbol{v},\boldsymbol{u}_h\rangle_{\mathbb{Q}_h} \tag{3.20}$$

3.2 Mixed Hyperelasticity

For hyperelastic material we are not in the linear strain range, thus we cannot use $p_{\text{hyd}} = -\frac{\text{trace}\,\boldsymbol{\sigma}}{3}$ in terms of small strain $\boldsymbol{\varepsilon}$. To define pressure, first, we need to derive the Cauchy stress

for hyperelastic which is [44]

$$\boldsymbol{\sigma} = \frac{1}{J}\boldsymbol{\tau} = \frac{1}{J}\boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^T \tag{3.21}$$

then from definition of general hyperelastic constitutive law (2.25) and (2.26), hydrostatic pressure is

$$\begin{aligned} p_{\text{hyd}} = -\frac{\operatorname{trace}\boldsymbol{\sigma}}{3} &= -\frac{1}{3J}\operatorname{trace}(\boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^T) = -\frac{1}{3J}\operatorname{trace}(\boldsymbol{S}\boldsymbol{F}^T\boldsymbol{F}) = -\frac{1}{3J}\operatorname{trace}(\boldsymbol{S}\boldsymbol{C}) \\ &= -\frac{1}{3J}\operatorname{trace}\left(\frac{\partial\psi_{\text{vol}}}{\partial J}\ J\boldsymbol{I} + \sum_{i=1}^{3}\frac{\partial\psi_{\text{iso}}}{\partial\bar{\mathbb{I}}_i}\frac{\partial\bar{\mathbb{I}}_i}{\partial\boldsymbol{E}}\boldsymbol{C}\right) \\ &= -\frac{\partial\psi_{\text{vol}}}{\partial J} = -k\frac{\partial U(J)}{\partial J}, \end{aligned} \tag{3.22}$$

where we have used $\operatorname{trace}(\boldsymbol{A}\boldsymbol{B}) = \operatorname{trace}(\boldsymbol{B}\boldsymbol{A})$ and

$$\begin{aligned} \operatorname{trace}\left(\frac{\partial\bar{\mathbb{I}}_1}{\partial\boldsymbol{E}}\boldsymbol{C}\right) &= 2J^{-2/3}\operatorname{trace}\left(\boldsymbol{C} - \frac{1}{3}\mathbb{I}_1\boldsymbol{I}\right) = 0, \\ \operatorname{trace}\left(\frac{\partial\bar{\mathbb{I}}_2}{\partial\boldsymbol{E}}\boldsymbol{C}\right) &= 2J^{-4/3}\operatorname{trace}\left(\mathbb{I}_1\boldsymbol{C} - \boldsymbol{C}^2 - \frac{2}{3}\mathbb{I}_2\boldsymbol{I}\right) = 0, \\ \operatorname{trace}\left(\frac{\partial\bar{\mathbb{I}}_3}{\partial\boldsymbol{E}}\boldsymbol{C}\right) &= 0, \end{aligned} \tag{3.23}$$

and considered the volumetric energy in the general form

$$\psi_{\text{vol}} = kU(J) = \frac{k}{4}\left(J^2 - 1 - 2\log J\right). \tag{3.24}$$

However, similar to mixed linear case (3.10), we define a pressure-like variable in terms of bulk and primal bulk moduli

$$p = -(k - k_p)\frac{\partial U}{\partial J} = p_{\text{hyd}} + k_p U', \tag{3.25}$$

and general mixed constitutive equation by substituting p_{hyd} in (2.29) to arrive at

$$\boldsymbol{S}(\boldsymbol{u}, p) = \boldsymbol{S}_{\text{iso}} + \underbrace{\left(k_p J U' - pJ\right)\boldsymbol{C}^{-1}}_{\boldsymbol{S}_{\text{vol}}}, \tag{3.26}$$

where $\boldsymbol{S}$ is the first Piola-Kirchhoff stress tensor. With (3.25) and (3.26), we can state the general mixed hyperelastic strong form as follow: Given body force $\rho_0\boldsymbol{g}$, Dirichlet boundary $\boldsymbol{g}^{\boldsymbol{u}}$ and applied

traction $\bar{\boldsymbol{t}}$, find the displacement and pressure-like variable $(\boldsymbol{u}, p) \in \mathbb{V} \times \mathbb{Q}$ such that:

$$\begin{aligned} -\nabla_X \cdot \boldsymbol{P} - \rho_0 \boldsymbol{g} &= 0. \qquad \text{in } \Omega_0 \\ -U' - \frac{p}{k - k_p} &= 0. \qquad \text{in } \Omega \\ \boldsymbol{u} &= \boldsymbol{g}^{\boldsymbol{u}}, \quad \text{on } \partial\Omega_0^D \\ \boldsymbol{P} \cdot \boldsymbol{N} &= \bar{\boldsymbol{t}}. \qquad \text{on } \partial\Omega_0^N \end{aligned} \tag{3.27}$$

and its weak formulation as:

$$\begin{aligned} R_u(\boldsymbol{u}, p) &\coloneqq \int_{\Omega_0} \nabla_X \boldsymbol{v} : \boldsymbol{P}\, dV - \int_{\Omega_0} \boldsymbol{v} \cdot \rho_0 \boldsymbol{g}\, dV - \int_{\partial\Omega_0^N} \boldsymbol{v} \cdot \bar{\boldsymbol{t}}\, dS = 0, \qquad \forall \in \boldsymbol{v} \in \mathbb{V}^0 \\ R_p(\boldsymbol{u}, p) &\coloneqq \int_{\Omega_0} q\, L\, J\, dV = 0, \qquad \forall \in q \in \mathbb{Q} \end{aligned} \tag{3.28}$$

where $(.)_0$ indicates the initial configuration, $\boldsymbol{P} = \boldsymbol{F}\boldsymbol{S}$ is the first Piola-Kirchhoff stress tensor, $\boldsymbol{N}$ is the unit normal vector, $_X$ in ∇_X indicates that the gradient is calculated with respect to the initial configuration, and

$$L = -U' - \frac{p}{k - k_p}.$$

Discretization of (3.28) produces a finite-dimensional system of nonlinear algebraic equations, which we solve using Newton-Raphson methods. One attractive feature of Galerkin discretization is that we can arrive at the same linear system by discretizing the Newton linearization of the continuous form; that is, discretization and differentiation (Newton linearization) commute. In general for solving a nonlinear equation $R(\boldsymbol{u}) = 0$, we use the first-order (Taylor's) expansion

$$0 = R(\boldsymbol{u}) + \mathrm{d}R(\boldsymbol{u}; \mathrm{d}\boldsymbol{u}) + o(\mathrm{d}\boldsymbol{u}), \tag{3.29}$$

where d and $\mathrm{d}\boldsymbol{u}$ denote the linearization operator and the increment of the displacement field $\boldsymbol{u}$. The reminder $o(\mathrm{d}\boldsymbol{u})$ is a small error that tends to zero faster than $\mathrm{d}\boldsymbol{u} \longrightarrow 0$. The second term $\mathrm{d}R$ is the linear change in R due to $\mathrm{d}\boldsymbol{u}$ at $\boldsymbol{u}$. In fact, it is the directional derivative of R at a given $\boldsymbol{u}$ (fixed) in the direction of the incremental displacement field

$$\mathrm{d}R(\boldsymbol{u}) = \left.\frac{\partial R(\boldsymbol{u} + \epsilon\, \mathrm{d}\boldsymbol{u})}{\partial \epsilon}\right|_{\epsilon=0}. \tag{3.30}$$

For instance, the deformation gradient linearization is

$$\begin{aligned}\mathrm{d}\boldsymbol{F} &= \left.\frac{\partial \boldsymbol{F}(\boldsymbol{u}+\epsilon\,\mathrm{d}\boldsymbol{u})}{\partial\epsilon}\right|_{\epsilon=0} = \left(\frac{\partial \boldsymbol{I}}{\partial\epsilon} + \frac{\partial(\partial\boldsymbol{u}/\partial\boldsymbol{X})}{\partial\epsilon} + \frac{\partial\epsilon}{\partial\epsilon}\frac{\partial(\partial\,\mathrm{d}\boldsymbol{u}/\partial\boldsymbol{X})}{\partial\epsilon} + \epsilon\frac{\partial(\partial\,\mathrm{d}\boldsymbol{u}/\partial\boldsymbol{X})}{\partial\epsilon}\right)\Bigg|_{\epsilon=0} \\ &= \frac{\partial\,\mathrm{d}\boldsymbol{u}}{\partial\boldsymbol{X}} = \nabla_X\,\mathrm{d}\boldsymbol{u}.\end{aligned} \tag{3.31}$$

To simplify the linearization of other quantities we can use (3.31). For example the linearization of Green-Lagrange strain is

$$\mathrm{d}\boldsymbol{E} = \frac{\partial \boldsymbol{E}}{\partial \boldsymbol{F}} : \mathrm{d}\boldsymbol{F} = \frac{1}{2}\left(\mathrm{d}\boldsymbol{F}^T\boldsymbol{F} + \boldsymbol{F}^T\,\mathrm{d}\boldsymbol{F}\right) = \mathrm{sym}\left(\boldsymbol{F}^T\,\mathrm{d}\boldsymbol{F}\right). \tag{3.32}$$

With above explanation, the linearized form of (3.28) (assume constant traction and body force) may be stated as: Find $(\mathrm{d}\boldsymbol{u}, \mathrm{d}p) \in \mathbb{V}\times\mathbb{Q}$ such that for all $(\boldsymbol{v}, q)\in\mathbb{V}_0\times\mathbb{Q}$

$$\begin{aligned}\int_{\Omega_0} \nabla_X\boldsymbol{v} : (\mathrm{d}\boldsymbol{F}\boldsymbol{S} + \boldsymbol{F}\,\mathrm{d}\boldsymbol{S})\; dV &= -R_u(\boldsymbol{u}, p), \\ \int_{\Omega_0} q\,(\mathrm{d}L\,J + L\;\mathrm{d}J)\; dV &= -R_p(\boldsymbol{u}, p),\end{aligned} \tag{3.33}$$

where

$$\mathrm{d}\boldsymbol{S} = \frac{\partial \boldsymbol{S}_{\mathrm{iso}}}{\partial\boldsymbol{E}} : \mathrm{d}\boldsymbol{E} + \frac{\partial \boldsymbol{S}_{\mathrm{vol}}}{\partial\boldsymbol{E}} : \mathrm{d}\boldsymbol{E} + \frac{\partial \boldsymbol{S}_{\mathrm{vol}}}{\partial p}\,\mathrm{d}p = \mathrm{d}\boldsymbol{S}_{\mathrm{iso}} + \mathrm{d}\boldsymbol{S}^u_{\mathrm{vol}} + \mathrm{d}\boldsymbol{S}^p_{\mathrm{vol}}, \tag{3.34}$$

For the Neo-Hookean model (see (2.33) and (3.26)), with second Piola-Kirchhoff stress

$$\boldsymbol{S} = \boldsymbol{S}_{\mathrm{iso}} + \boldsymbol{S}_{\mathrm{vol}} = 2\mu J^{-2/3}\boldsymbol{C}^{-1}\boldsymbol{E}_{\mathrm{dev}} + \left(k_p JU' - pJ\right)\boldsymbol{C}^{-1}, \tag{3.35}$$

we can derive its linearization as

$$\begin{aligned}\mathrm{d}\boldsymbol{S}_{\mathrm{iso}} &= -\frac{2}{3}\mu J^{-2/3}(\boldsymbol{C}^{-1} : \mathrm{d}\boldsymbol{E})\left(\boldsymbol{I}_3 - \frac{1}{3}\mathbb{I}_1\boldsymbol{C}^{-1}\right) \\ &\quad - \frac{1}{3}\mu J^{-2/3}\left(2\,\mathrm{trace}\,\mathrm{d}\boldsymbol{E}\,\boldsymbol{C}^{-1} + \mathbb{I}_1\,\mathrm{d}\boldsymbol{C}^{-1}\right), \\ &= -\frac{4}{3}\mu J^{-2/3}(\boldsymbol{C}^{-1} : \mathrm{d}\boldsymbol{E})\boldsymbol{C}^{-1}\boldsymbol{E}_{\mathrm{dev}} - \frac{1}{3}\mu J^{-2/3}\left(2\,\mathrm{trace}\,\mathrm{d}\boldsymbol{E}\,\boldsymbol{C}^{-1} + \mathbb{I}_1\,\mathrm{d}\boldsymbol{C}^{-1}\right)\end{aligned} \tag{3.36}$$

and

$$\begin{aligned}\mathrm{d}\boldsymbol{S}^u_{\mathrm{vol}} &= \left(k_p\,\mathrm{d}J\,U' + k_p JU''\,\mathrm{d}J - p\,\mathrm{d}J\right)\boldsymbol{C}^{-1} + \left(k_p JU' - pJ\right)\mathrm{d}\boldsymbol{C}^{-1}, \\ &= \left[k_p J^2U'' + k_p JU' - pJ\right]\left(\boldsymbol{C}^{-1} : \mathrm{d}\boldsymbol{E}\right)\boldsymbol{C}^{-1} + \left(k_p JU' - pJ\right)\mathrm{d}\boldsymbol{C}^{-1}, \\ \mathrm{d}\boldsymbol{S}^p_{\mathrm{vol}} &= -\,\mathrm{d}pJ\boldsymbol{C}^{-1},\end{aligned} \tag{3.37}$$

where we have used

$$\mathrm{d}J = \frac{\partial J}{\partial \boldsymbol{E}} : \mathrm{d}\boldsymbol{E} = J\boldsymbol{C}^{-1} : \mathrm{d}\boldsymbol{E}, \tag{3.38}$$

and

$$\mathrm{d}\boldsymbol{C}^{-1} = \frac{\partial \boldsymbol{C}^{-1}}{\partial \boldsymbol{E}} : \mathrm{d}\boldsymbol{E} = -2\boldsymbol{C}^{-1}\,\mathrm{d}\boldsymbol{E}\,\boldsymbol{C}^{-1}. \tag{3.39}$$

The second equation of (3.33) also can be simplified as

$$\begin{aligned} \mathrm{d}LJ + L\,\mathrm{d}J &= \left(-U^{''}\,\mathrm{d}J - \frac{\mathrm{d}p}{k-k_p}\right) J + L\,\mathrm{d}J = \left(-JU^{''} + L\right)\mathrm{d}J - \frac{J\,\mathrm{d}p}{k-k_p}, \\ &= -\left(J^2U^{''} + JU^{'} + \frac{Jp}{k-k_p}\right)\boldsymbol{C}^{-1} : \mathrm{d}\boldsymbol{E} - \frac{J\,\mathrm{d}p}{k-k_p}. \end{aligned} \tag{3.40}$$

Note that for different material models like Mooney-Rivlin and Ogden, the volumetric stress and the second equation of (3.33) and their linearization (3.37), (3.40) are the same, and we only need to derive the isochoric part, which are given in Appendix C.

Keeping above linearization in mind, we can rewrite (3.33) in short-hand notation as

$$\begin{bmatrix} \mathcal{A} & \mathcal{B} \\ \mathcal{C} & \mathcal{D} \end{bmatrix} \begin{bmatrix} \mathrm{d}\boldsymbol{u} \\ \mathrm{d}p \end{bmatrix} = - \begin{bmatrix} R_u \\ R_p \end{bmatrix} \tag{3.41}$$

where R_u, R_p are defined in (3.28), and the block operators are

$$\begin{aligned} \mathcal{A} &\sim \int_{\Omega_0} \nabla_X \boldsymbol{v} : (\mathrm{d}\boldsymbol{F}\boldsymbol{S} + \boldsymbol{F}\,\mathrm{d}\boldsymbol{S}_{\mathrm{iso}} + \boldsymbol{F}\,\mathrm{d}\boldsymbol{S}^u_{\mathrm{vol}})\; dV, \\ \mathcal{B} &\sim \int_{\Omega_0} \nabla_X \boldsymbol{v} : \boldsymbol{F}\,\mathrm{d}\boldsymbol{S}^p_{\mathrm{vol}}\, dV = -\int_{\Omega_0} \nabla_X \boldsymbol{v} : \mathrm{d}p J\boldsymbol{F}^{-T}\, dV \\ \mathcal{C} &\sim -\int_{\Omega_0} q\left(J^2U^{''} + JU^{'} + \frac{Jp}{k-k_p}\right)\boldsymbol{C}^{-1} : \mathrm{d}\boldsymbol{E}\, dV, \\ \mathcal{D} &\sim -\int_{\Omega_0} q\frac{J\,\mathrm{d}p}{k-k_p}\, dV. \end{aligned} \tag{3.42}$$

The block structure matrix in (3.41) is not symmetric [22]. However, we can write the mixed formulation for incompressible materials by the variational approach as discussed in the next section.

It should be noted that for displacement-based formulation (single field), we only solve the first equation of (3.33) with

$$\boldsymbol{S} = \boldsymbol{S}_{\text{iso}} + \boldsymbol{S}_{\text{vol}} = 2\mu J^{-2/3}\boldsymbol{C}^{-1}\boldsymbol{E}_{\text{dev}} + kJU'\,\boldsymbol{C}^{-1}, \tag{3.43}$$

where the linearization of the isochoric part is the same as (3.36), and

$$\begin{aligned} \mathrm{d}\boldsymbol{S}^{u}_{\text{vol}} &= \left[kJ^2U'' + kJU'\right]\left(\boldsymbol{C}^{-1}\!:\mathrm{d}\boldsymbol{E}\right)\boldsymbol{C}^{-1} + \left(kJU'\right)\mathrm{d}\boldsymbol{C}^{-1}, \\ \mathrm{d}\boldsymbol{S}^{p}_{\text{vol}} &= 0. \end{aligned} \tag{3.44}$$

3.3 Perturbed Lagrange-multiplier method

In the previous sections section 3.1 and section 3.2, we introduced the strong form of general mixed formulation for small and finite strain and by considering test functions $(\boldsymbol{v}, q)$ as derived in (3.12) and (3.28). Alternatively, we can derive the mixed $\boldsymbol{u} - p$ weak formulation based on minimization of two fields functional $\Pi(\boldsymbol{u}, p)$, which it is known as the perturbed Lagrangian approach. For the mixed linear case, we can write

$$\begin{aligned} \Pi(\boldsymbol{u}, p) &= \int_\Omega \left[\mu\,\boldsymbol{\varepsilon}_{\text{dev}}\!:\boldsymbol{\varepsilon}_{\text{dev}} - p\,\operatorname{trace}\boldsymbol{\varepsilon} + \frac{k_p}{2}\left(\operatorname{trace}\boldsymbol{\varepsilon}\right)^2 - \frac{p^2}{2(k-k_p)}\right] dv - \Pi_{\text{ext}}(\boldsymbol{u}) \\ \Pi_{\text{ext}}(\boldsymbol{u}) &= \int_\Omega \boldsymbol{u}\cdot\rho\boldsymbol{g}\,dv + \int_{\partial\Omega}\boldsymbol{u}\cdot\bar{\boldsymbol{t}}\,ds \end{aligned} \tag{3.45}$$

and by invoking the stationarity of Π with respect to $\boldsymbol{u}$ and p, we obtain

$$\begin{aligned} \int_\Omega \nabla\delta\boldsymbol{u}\!:\left[2\mu\,\boldsymbol{\varepsilon}_{\text{dev}} + \left(k_p\operatorname{trace}\boldsymbol{\varepsilon} - p\right)\boldsymbol{I}\right] dv - \boldsymbol{L}_{\text{ext}}(\delta\boldsymbol{u}) = 0, \\ \int_\Omega \delta p\left(-\operatorname{trace}\boldsymbol{\varepsilon} - \frac{p}{k-k_p}\right) dv = 0, \end{aligned} \tag{3.46}$$

where $\boldsymbol{L}_{\text{ext}}(\delta\boldsymbol{u}) = \int_\Omega \delta\boldsymbol{u}\cdot\rho\boldsymbol{g}\,dv + \int_{\partial\Omega}\delta\boldsymbol{u}\cdot\bar{\boldsymbol{t}}\,ds$, and we have used

$$\begin{aligned} \frac{\partial\operatorname{trace}\boldsymbol{\varepsilon}}{\partial\boldsymbol{u}}\delta\boldsymbol{u} = \frac{\partial\operatorname{trace}\boldsymbol{\varepsilon}}{\partial\boldsymbol{\varepsilon}}\frac{\partial\boldsymbol{\varepsilon}}{\partial\boldsymbol{u}}\delta\boldsymbol{u} = \boldsymbol{I}\!:\delta\boldsymbol{\varepsilon}, \\ \delta\boldsymbol{\varepsilon} = \frac{1}{2}\left(\nabla\delta\boldsymbol{u} + \nabla\delta\boldsymbol{u}^T\right) = \operatorname{sym}(\nabla\delta\boldsymbol{u}), \end{aligned} \tag{3.47}$$

in which $\delta\boldsymbol{u}$ and δp are virtual displacement and pressure and can be seen as the test functions $\boldsymbol{v}, q$, i.e., $\nabla\delta\boldsymbol{u} = \nabla\boldsymbol{v}$, $\delta p = q$. It is clear that the weak form (3.46) agrees with what we obtained in (3.12).

However, the hyperelastic weak form (3.28), can not be derived by minimizing any functional [22] and as shown its linearization is not symmetric.

To write the mixed functional $\Pi(\boldsymbol{u}, p)$ for hyperelastic, we need to consider the strain energy of the form [22]

$$\psi = \psi_{\text{vol}}(J) + \psi_{\text{iso}}(\bar{\boldsymbol{C}}) = \frac{k}{2}\left(\hat{U}(J)\right)^2 + \psi_{\text{iso}}(\bar{\boldsymbol{C}}), \tag{3.48}$$

where the function $\hat{U}$ requires condition

$$\hat{U}(J) = 0 \quad \text{if and only if} \quad J = 1,$$

which by comparing the volumetric part of above equation with (3.24), we realize that if

$$U(J) = \frac{1}{2}\hat{U}^2(J), \tag{3.49}$$

the strain energy will be the same for both methods. Using the quadratic form of volumetric energy, we can write the general mixed functional for hyperelasticity in terms of bulk moduli k, k_p as

$$\begin{aligned}
\Pi(\boldsymbol{u}, p) &= \int_{\Omega_0} \left[\psi_{\text{iso}}(\bar{\boldsymbol{C}}) - p\,\hat{U}(J) + \frac{k_p}{2}\left(\hat{U}(J)\right)^2 - \frac{p^2}{2(k-k_p)}\right] dV - \Pi_{\text{ext}}(\boldsymbol{u}), \\
\Pi_{\text{ext}}(\boldsymbol{u}) &= \int_{\Omega_0} \boldsymbol{u}\cdot\rho_0\boldsymbol{g}\, dV + \int_{\partial\Omega_0} \boldsymbol{u}\cdot\bar{\boldsymbol{t}}\, dS.
\end{aligned} \tag{3.50}$$

Finding the stationary conditions with respect to $\delta\boldsymbol{u} = \boldsymbol{v}$ and $\delta p = q$ by taking Gateaux derivative gives the weak form

$$\begin{aligned}
R_u^{PL}(\boldsymbol{u}, p) &\coloneqq \int_{\Omega_0} \boldsymbol{F}\left[\boldsymbol{S}_{\text{iso}} + \underbrace{\left(k_p\hat{U} - p\right) J\,\hat{U}'\,\boldsymbol{C}^{-1}}_{\hat{\boldsymbol{S}}_{\text{vol}}}\right] : \nabla_X \boldsymbol{v}\, dV - L_{\text{ext}}(\boldsymbol{v}) = 0 \\
R_p^{PL}(\boldsymbol{u}, p) &\coloneqq \int_{\Omega_0} \left(-\hat{U}(J) - \frac{p}{k-k_p}\right) q\, dV = 0
\end{aligned} \tag{3.51}$$

and its linearization can be derived as

$$\begin{aligned}
\int_{\Omega_0} \nabla_X \boldsymbol{v} : (\boldsymbol{F}\,\mathrm{d}\boldsymbol{S} + \mathrm{d}\boldsymbol{F}\boldsymbol{S})\, dV &= -R_u^{PL}(\boldsymbol{u}, p), \\
\int_{\Omega_0} q\left(-\,\mathrm{d}\hat{U} - \frac{\mathrm{d}p}{k-k_p}\right) dV &= -R_p^{PL}(\boldsymbol{u}, p),
\end{aligned} \tag{3.52}$$

where $\mathrm{d}\hat{U} = \hat{U}'\,\mathrm{d}J$, and $\mathrm{d}\boldsymbol{S} = \mathrm{d}\boldsymbol{S}_{\mathrm{iso}} + \mathrm{d}\hat{\boldsymbol{S}}^{u}_{\mathrm{vol}} + \mathrm{d}\hat{\boldsymbol{S}}^{p}_{\mathrm{vol}}$.

In perturbed Lagrange-multiplier method, the isochoric stress $\boldsymbol{S}_{\mathrm{iso}}$ and its linearization $\mathrm{d}\boldsymbol{S}_{iso}$ are the same as given in (3.35) and (3.36) but the volumetric part is different i.e.,

$$\hat{\boldsymbol{S}}_{\mathrm{vol}} = \Big(k_p\hat{U} - p\Big)\, J\,\hat{U}'\,\boldsymbol{C}^{-1} \tag{3.53}$$

with its linearization

$$\begin{aligned}
\mathrm{d}\hat{\boldsymbol{S}}^{u}_{\mathrm{vol}} &= \Big[k_p(J\hat{U}')^2 + \Big(k_p\hat{U} - p\Big)\Big(J\hat{U}' + J^2\hat{U}''\Big)\,(\boldsymbol{C}^{-1}\!:\mathrm{d}\boldsymbol{E})\Big]\,\boldsymbol{C}^{-1} + \Big(k_p\hat{U} - p\Big)\,J\hat{U}'\,\mathrm{d}\boldsymbol{C}^{-1},\\
\mathrm{d}\hat{\boldsymbol{S}}^{p}_{\mathrm{vol}} &= -\,\mathrm{d}p J\hat{U}'\boldsymbol{C}^{-1}.
\end{aligned} \tag{3.54}$$

Furthermore, unlike (3.41), the block structure system

$$\begin{bmatrix} \mathcal{A} & \mathcal{B} \\ \mathcal{C} & \mathcal{D} \end{bmatrix} \begin{bmatrix} \mathrm{d}\boldsymbol{u} \\ \mathrm{d}p \end{bmatrix} = - \begin{bmatrix} R^{PL}_{u} \\ R^{PL}_{p} \end{bmatrix} \tag{3.55}$$

obtained with PL approach is symmetric i.e., $\mathcal{C} = \mathcal{B}^T$, where R^{PL}_{u}, R^{PL}_{p} are defined in (3.51), and the block operators are

$$\begin{aligned}
\mathcal{A} &\sim \int_{\Omega_0} \nabla_X \boldsymbol{v} : \Big(\mathrm{d}\boldsymbol{F}\boldsymbol{S} + \boldsymbol{F}\,\mathrm{d}\boldsymbol{S}_{\mathrm{iso}} + \boldsymbol{F}\,\mathrm{d}\hat{\boldsymbol{S}}^{u}_{\mathrm{vol}}\Big)\, dV,\\
\mathcal{B} &\sim \int_{\Omega_0} \nabla_X \boldsymbol{v} : \boldsymbol{F}\,\mathrm{d}\hat{\boldsymbol{S}}^{p}_{\mathrm{vol}}\, dV = -\int_{\Omega_0} \nabla_X \boldsymbol{v} : \mathrm{d}p\, J\hat{U}'\boldsymbol{F}^{-T}\, dV\\
\mathcal{C} &\sim -\int_{\Omega_0} q\, J\hat{U}'\boldsymbol{C}^{-1} : \mathrm{d}\boldsymbol{E}\, dV = -\int_{\Omega_0} q\, J\hat{U}'\boldsymbol{F}^{-T} : \nabla_X\, \mathrm{d}\boldsymbol{u}\, dV \equiv \mathcal{B}^T,\\
\mathcal{D} &\sim -\int_{\Omega_0} q\frac{\mathrm{d}p}{k - k_p}\, dV.
\end{aligned} \tag{3.56}$$

3.4 Current Configuration

In the preceding sections, all equations have been formulated in the initial configuration. This may feel convenient in that the computational domain is clearly independent of the solution, but there are some advantages to defining the equations in the current configuration.

- Body forces (such as gravity), traction, and contact are more easily defined in the current configuration.

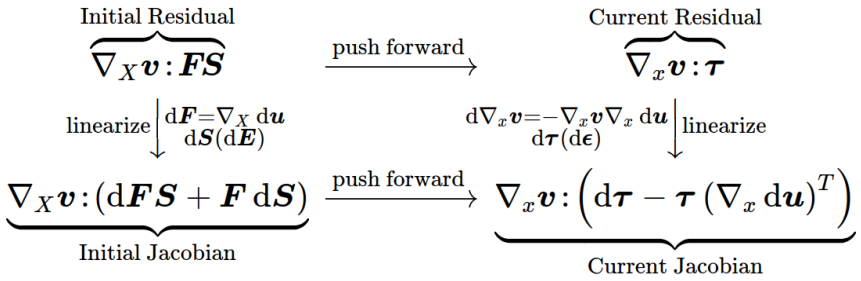


Figure 3.1: Residual and Jacobian in initial and current configurations commute in multiple paths.

- Mesh quality in the initial configuration can be very bad for large deformation.
- The required storage and numerical representation can be smaller in the current configuration.

Most of the benefit in last item can be attained solely by moving the Jacobian representation to the current configuration [29], though residual evaluation may also be slightly faster in current configuration. There are multiple commuting paths from the nonlinear weak form in initial configuration to the Jacobian weak form in current configuration. One may push forward to the current configuration and then linearize or linearize in initial configuration and then push forward, as summarized below in Figure 3.1.

The first term of first equation in (3.28) can be rewritten in terms of the symmetric Kirchhoff stress tensor $\boldsymbol{\tau} = J\boldsymbol{\sigma} = \boldsymbol{P}\boldsymbol{F}^T = \boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^T$ as

$$\nabla_X \boldsymbol{v} : \boldsymbol{P} = \nabla_X \boldsymbol{v} : \boldsymbol{\tau}\boldsymbol{F}^{-T} = \nabla_X \boldsymbol{v}\boldsymbol{F}^{-1} : \boldsymbol{\tau} = \nabla_x \boldsymbol{v} : \boldsymbol{\tau}$$

therefore, the weak form in terms of $\boldsymbol{\tau}$ and gradient with respect to current configuration ∇_x is

$$\begin{aligned} r_u(\boldsymbol{u}, p) &\coloneqq \int_{\Omega_0} \nabla_x \boldsymbol{v} : \boldsymbol{\tau}\, dV - \int_{\Omega_0} \boldsymbol{v} \cdot \rho_0 \boldsymbol{g}\, dV - \int_{\partial\Omega_0^N} \boldsymbol{v} \cdot \bar{\boldsymbol{t}}\, dS = 0, \qquad \forall \in \boldsymbol{v} \in \mathbb{V}^0 \\ r_p(\boldsymbol{u}, p) &\coloneqq \int_{\Omega_0} q\, L\, J\, dV = 0, \qquad \forall \in q \in \mathbb{Q} \end{aligned} \tag{3.57}$$

Similarly we can rewrite the first equation of the weak form derived by variational approach (3.51) in current configuration.

To derive a Newton linearization of (3.57), first we define

$$\nabla_x \,\mathrm{d}\boldsymbol{u} = \nabla_X \,\mathrm{d}\boldsymbol{u}\ \boldsymbol{F}^{-1} = \mathrm{d}\boldsymbol{F}\boldsymbol{F}^{-1}, \tag{3.58}$$

then by expanding the directional derivative of $\nabla_x \boldsymbol{v} : \boldsymbol{\tau}$, we arrive at

$$\mathrm{d}\ (\nabla_x \boldsymbol{v} : \boldsymbol{\tau}) = \mathrm{d}\ (\nabla_x \boldsymbol{v}) : \boldsymbol{\tau} + \nabla_x \boldsymbol{v} : \mathrm{d}\boldsymbol{\tau}. \tag{3.59}$$

The first term of (3.59) can be written as

$$\begin{aligned} \mathrm{d}\,(\nabla_x \boldsymbol{v}) : \boldsymbol{\tau} = \mathrm{d}\,\left(\nabla_X \boldsymbol{v}\boldsymbol{F}^{-1}\right) : \boldsymbol{\tau} &= \left(\underbrace{\nabla_X\,(\mathrm{d}\boldsymbol{v})}_{0}\, \boldsymbol{F}^{-1} + \nabla_X \boldsymbol{v}\, \mathrm{d}\boldsymbol{F}^{-1} \right) : \boldsymbol{\tau} \\ &= \left(-\nabla_X \boldsymbol{v}\boldsymbol{F}^{-1}\, \mathrm{d}\boldsymbol{F}\boldsymbol{F}^{-1}\right) : \boldsymbol{\tau} = \left(-\nabla_x \boldsymbol{v}\, \mathrm{d}\boldsymbol{F}\boldsymbol{F}^{-1}\right) : \boldsymbol{\tau} \\ &= \left(-\nabla_x \boldsymbol{v} \nabla_x\, \mathrm{d}\boldsymbol{u}\right) : \boldsymbol{\tau} = -\nabla_x \boldsymbol{v} : \boldsymbol{\tau}\, (\nabla_x\, \mathrm{d}\boldsymbol{u})^T \end{aligned} \tag{3.60}$$

where we have used $\mathrm{d}\boldsymbol{F}^{-1} = -\boldsymbol{F}^{-1}\,\mathrm{d}\boldsymbol{F}\boldsymbol{F}^{-1}$ and (3.58). By making use of (3.59) and (3.60) in (3.57), we arrive at the Jacobian in the current configuration: Find $(\mathrm{d}\boldsymbol{u}, \mathrm{d}p) \in \mathbb{V} \times \mathbb{Q}$ such that for all $(\boldsymbol{v}, q) \in \mathbb{V}_0 \times \mathbb{Q}$

$$\begin{aligned} \int_{\Omega_0} \nabla_x \boldsymbol{v} : \left(\mathrm{d}\boldsymbol{\tau} - \boldsymbol{\tau}\, (\nabla_x\, \mathrm{d}\boldsymbol{u})^T\right)\, dV &= -r_u(\boldsymbol{u}, p), \\ \int_{\Omega_0} q\, (\mathrm{d}L\, J + L\ \mathrm{d}J)\, dV &= -r_p(\boldsymbol{u}, p), \end{aligned} \tag{3.61}$$

Alternatively, we can transform the jacobian in initial configuration to current via

$$\nabla_X \boldsymbol{v} : \mathrm{d}\boldsymbol{P} = (\nabla_X \boldsymbol{v})\, \boldsymbol{F}^{-1} : \mathrm{d}\boldsymbol{P}\boldsymbol{F}^T = \nabla_x \boldsymbol{v} : \mathrm{d}\boldsymbol{P}\boldsymbol{F}^T = \nabla_x \boldsymbol{v} : \left(\mathrm{d}\boldsymbol{\tau} - \boldsymbol{\tau}\, (\nabla_x\, \mathrm{d}\boldsymbol{u})^T\right)$$

and with existing derivation of $\mathrm{d}\boldsymbol{S}$ in Appendix C, we can compute

$$\begin{aligned} \mathrm{d}\boldsymbol{\tau} - \boldsymbol{\tau}\, (\nabla_x\, \mathrm{d}\boldsymbol{u})^T = \mathrm{d}\boldsymbol{P}\boldsymbol{F}^T &= \mathrm{d}\boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^T + \boldsymbol{F}\, \mathrm{d}\boldsymbol{S}\boldsymbol{F}^T \\ &= \mathrm{d}\boldsymbol{F}\boldsymbol{F}^{-1}\boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^T + \boldsymbol{F}\, \mathrm{d}\boldsymbol{S}\boldsymbol{F}^T \\ &= \nabla_x\, \mathrm{d}\boldsymbol{u}\, \boldsymbol{\tau} + \boldsymbol{F}\, \mathrm{d}\boldsymbol{S}\boldsymbol{F}^T \end{aligned} \tag{3.62}$$

For the Neo-Hookean model (3.35) we can derive (see (2.35))

$$\boldsymbol{\tau} = \boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^T = \boldsymbol{\tau}_{\mathrm{iso}} + \boldsymbol{\tau}_{\mathrm{vol}} = 2\mu J^{-2/3} \boldsymbol{e}_{\mathrm{dev}} + \left(k_p J U' - pJ\right) \boldsymbol{I}, \tag{3.63}$$

where $\boldsymbol{e}_{\text{dev}}$ is the deviatoric part of the Green-Euler tensor (see Appendix A), and we have used

$$\boldsymbol{F}\boldsymbol{C}^{-1}\boldsymbol{E}_{\text{dev}}\boldsymbol{F}^{T} = \boldsymbol{F}^{-T}\left(\boldsymbol{E} - \frac{1}{3}\mathbb{I}_1(\boldsymbol{E})\boldsymbol{I}\right)\boldsymbol{F}^{T} = \boldsymbol{e}_{\text{dev}}. \tag{3.64}$$

To compute $\boldsymbol{F}\,\mathrm{d}\boldsymbol{S}\boldsymbol{F}^{T}$, first we start by (3.36)

$$\boldsymbol{F}\,\mathrm{d}\boldsymbol{S}_{\text{iso}}\boldsymbol{F}^{T} = -\frac{2}{3}\mu J^{-2/3}\left(2\,\text{trace}(\mathrm{d}\boldsymbol{\epsilon})\boldsymbol{e}_{\text{dev}} + \text{trace}(\mathrm{d}\boldsymbol{e})\boldsymbol{I} - \mathbb{I}_1\,\mathrm{d}\boldsymbol{\epsilon}\right), \tag{3.65}$$

where $\mathbb{I}_1 = \text{trace}(\boldsymbol{b})$, $\boldsymbol{b}$ is the left Cauchy-Green tensor,

$$\mathrm{d}\boldsymbol{\epsilon} \equiv \frac{1}{2}\left(\nabla_x\,\mathrm{d}\boldsymbol{u} + (\nabla_x\,\mathrm{d}\boldsymbol{u})^{T}\right) = \text{sym}\left(\nabla_x\,\mathrm{d}\boldsymbol{u}\right), \tag{3.66}$$

and we have used the following derivations to derive the equivalent terms in current configurations:

$$\text{trace}\left(\mathrm{d}\boldsymbol{E}\right) = \text{trace}\left(\boldsymbol{F}^{T}\,\mathrm{d}\boldsymbol{F}\right) = \text{trace}\left(\boldsymbol{F}\,\mathrm{d}\boldsymbol{F}^{T}\right) = \text{trace}\left(\mathrm{d}\boldsymbol{F}\boldsymbol{F}^{T}\right) = \text{trace}\left(\mathrm{d}\boldsymbol{e}\right). \tag{3.67}$$

$$\begin{aligned}
\boldsymbol{C}^{-1}\!:\mathrm{d}\boldsymbol{E} &= \boldsymbol{F}^{-1}\boldsymbol{F}^{-T}\!:\boldsymbol{F}^{T}\,\mathrm{d}\boldsymbol{F} \\
&= \text{trace}(\boldsymbol{F}^{-1}\boldsymbol{F}^{-T}\boldsymbol{F}^{T}\,\mathrm{d}\boldsymbol{F}) \\
&= \text{trace}(\boldsymbol{F}^{-1}\,\mathrm{d}\boldsymbol{F}) \\
&= \text{trace}(\mathrm{d}\boldsymbol{F}\boldsymbol{F}^{-1}) = \text{trace}(\nabla_x\,\mathrm{d}\boldsymbol{u}) = \text{trace}\,\mathrm{d}\boldsymbol{\epsilon},
\end{aligned} \tag{3.68}$$

$$\boldsymbol{F}\,\mathrm{d}\boldsymbol{C}^{-1}\boldsymbol{F}^{T} = -2\boldsymbol{F}\boldsymbol{C}^{-1}\,\mathrm{d}\boldsymbol{E}\,\boldsymbol{C}^{-1}\boldsymbol{F}^{T} = -2\boldsymbol{F}^{-T}\,\mathrm{d}\boldsymbol{E}\,\boldsymbol{F}^{-1} = -2\,\mathrm{d}\boldsymbol{\epsilon}, \tag{3.69}$$

where $\boldsymbol{F}^{-T}\,\mathrm{d}\boldsymbol{E}\,\boldsymbol{F}^{-1}$ is simplified as

$$\begin{aligned}
\boldsymbol{F}^{-T}\,\mathrm{d}\boldsymbol{E}\,\boldsymbol{F}^{-1} &= \frac{1}{2}\boldsymbol{F}^{-T}\left(\boldsymbol{F}^{T}\,\mathrm{d}\boldsymbol{F} + \mathrm{d}\boldsymbol{F}^{T}\boldsymbol{F}\right)\boldsymbol{F}^{-1} \\
&= \frac{1}{2}\left(\mathrm{d}\boldsymbol{F}\boldsymbol{F}^{-1} + \boldsymbol{F}^{-T}\,\mathrm{d}\boldsymbol{F}^{T}\right) \\
&= \frac{1}{2}\Big(\nabla_x\,\mathrm{d}\boldsymbol{u} + (\nabla_x\,\mathrm{d}\boldsymbol{u})^{T}\Big) \equiv \mathrm{d}\boldsymbol{\epsilon}.
\end{aligned} \tag{3.70}$$

Then, from (3.37) we have

$$\begin{aligned}
\boldsymbol{F}\,\mathrm{d}\boldsymbol{S}^{u}_{\text{vol}}\boldsymbol{F}^{T} &= \Big[k_p J^2 U'' + k_p J U' - pJ\Big]\left(\text{trace}\,\mathrm{d}\boldsymbol{\epsilon}\right)\boldsymbol{I} - 2\left(k_p J U' - pJ\right)\mathrm{d}\boldsymbol{\epsilon}, \\
\boldsymbol{F}\,\mathrm{d}\boldsymbol{S}^{p}_{\text{vol}}\boldsymbol{F}^{T} &= -\,\mathrm{d}pJ\boldsymbol{I},
\end{aligned} \tag{3.71}$$

As you can see in current configuration we do not need to compute the inverse of a matrix like $\boldsymbol{C}^{-1}$ since it is replaced by identity matrix. Moreover, for computing $\boldsymbol{C}^{-1} : \mathrm{d}\boldsymbol{E}$ in initial configuration, we need 15 flops while this term in current configuration is substituted by $\operatorname{trace} \mathrm{d}\boldsymbol{\epsilon} = \operatorname{trace}(\nabla_x \, \mathrm{d}\boldsymbol{u})$, which requires 2 flops for its computation. Furthermore, the linearization of $\boldsymbol{C}^{-1}$ (3.39), which needs computing inverse of right Cauchy-Green tensor and two matrix-matrix multiplication is replaced by $-2\,\mathrm{d}\boldsymbol{\epsilon}$ term.

With above derivation for current configuration, we can rewrite (3.61) in a matrix form as

$$\begin{bmatrix} \mathcal{A} & \mathcal{B} \\ \mathcal{C} & \mathcal{D} \end{bmatrix} \begin{bmatrix} \mathrm{d}\boldsymbol{u} \\ \mathrm{d}p \end{bmatrix} = - \begin{bmatrix} r_u \\ r_p \end{bmatrix} \tag{3.72}$$

where r_u, r_p are defined in (3.57), and the block operators are

$$\begin{aligned}
\mathcal{A} &\sim \int_{\Omega_0} \nabla_x \boldsymbol{v} : \left(\nabla_x \, \mathrm{d}\boldsymbol{u} \, \boldsymbol{\tau} + \boldsymbol{F} \, \mathrm{d}\boldsymbol{S}_{\mathrm{iso}} \boldsymbol{F}^T + \boldsymbol{F} \, \mathrm{d}\boldsymbol{S}^u_{\mathrm{vol}} \boldsymbol{F}^T \right) \, dV, \\
\mathcal{B} &\sim \int_{\Omega_0} \nabla_x \boldsymbol{v} : \boldsymbol{F} \, \mathrm{d}\boldsymbol{S}^p_{\mathrm{vol}} \boldsymbol{F}^T \, dV = - \int_{\Omega_0} \nabla_x \boldsymbol{v} : \mathrm{d}p J \boldsymbol{I} \, dV = - \int_{\Omega_0} \operatorname{trace} \left(\nabla_x \boldsymbol{v} \right) \mathrm{d}p J \, dV \\
\mathcal{C} &\sim - \int_{\Omega_0} q \left(J^2 U'' + J U' + \frac{Jp}{k - k_p} \right) \operatorname{trace} \mathrm{d}\boldsymbol{\epsilon} \, dV, \\
\mathcal{D} &\sim - \int_{\Omega_0} q \frac{J \, \mathrm{d}p}{k - k_p} \, dV.
\end{aligned} \tag{3.73}$$

It should be noted that the perturbed Lagrange-multiplier formulation (3.55) can be transformed into current configuration similarly. Also for different material models, we derived the Kirchhoff stress $\boldsymbol{\tau}$ in subsection 2.2.2, and their linearization $\boldsymbol{F} \, \mathrm{d}\boldsymbol{S} \boldsymbol{F}^T$ are given in Appendix D.

3.4.1 Pressure Boundary Condition

One of the important load case is the pressure boundary loading which is caused by liquids or gases on the surface of the solid structure. The pressure boundary load depends upon the current state of the deformation and can be considered as a traction vector $\bar{\boldsymbol{t}} = -p_0 \boldsymbol{n}$ per unit current surface, acting in the direction of the outward unit normal $\boldsymbol{n}$. Therefore, the following surface

integral will be add to the right-hand-side of the first equation of (3.28) or (3.57)

$$\int_{\partial\Omega} \boldsymbol{v} \cdot (-p_0 \boldsymbol{n})\, ds, \tag{3.74}$$

where the normal on current surface is computed by

$$\boldsymbol{n} = \frac{\frac{\partial \boldsymbol{x}}{\partial \xi_1} \times \frac{\partial \boldsymbol{x}}{\partial \xi_2}}{|\frac{\partial \boldsymbol{x}}{\partial \xi_1} \times \frac{\partial \boldsymbol{x}}{\partial \xi_2}|} \tag{3.75}$$

in which $\xi_1, \xi_2 \in [-1, 1]^2$ are reference coordinate system on the face and $\boldsymbol{x}$ is the position vector in current configuration. If we write the surface area in terms of reference coordinate as $ds = |\frac{\partial \boldsymbol{x}}{\partial \xi_1} \times \frac{\partial \boldsymbol{x}}{\partial \xi_2}| d\xi_1 d\xi_2$, the pressure load (3.74) can be simplified to

$$-\int_{[-1,1]^2} \boldsymbol{v} \cdot p_0 \left(\frac{\partial \boldsymbol{x}}{\partial \xi_1} \times \frac{\partial \boldsymbol{x}}{\partial \xi_2} \right) d\xi_1 d\xi_2 \tag{3.76}$$

In deriving (3.33) we assume the traction is constant. However, in the case of pressure loading on boundary even for constant pressure, due to appearance of current state $\boldsymbol{x}$ the linearization of (3.76) is not zero and given by

$$-p_0 \int_{[-1,1]^2} \boldsymbol{v} \cdot \left(\frac{\partial\, \mathrm{d}\boldsymbol{u}}{\partial \xi_1} \times \frac{\partial \boldsymbol{x}}{\partial \xi_2} + \frac{\partial \boldsymbol{x}}{\partial \xi_1} \times \frac{\partial\, \mathrm{d}\boldsymbol{u}}{\partial \xi_2} \right) d\xi_1 d\xi_2. \tag{3.77}$$

3.5 Matrix-free formulation

Historically, conventional high-order finite element methods were rarely used for industrial problems because the Jacobian rapidly loses sparsity as the order is increased, leading to unaffordable solve times and memory requirements [24]. Matrix-free methods describe algorithms that do not require the explicit storage of a matrix in order to solve the algebraic system of equations resulting from the discretization of a PDE. The main motivation for matrix-free methods is to enable the solution of larger problems due to reduced memory requirements, but more importantly, to solve a problem faster due to a significant reduction in the number of operations and in the transfer of data from main memory.

In the context of finite element methods, matrix-free methods are typically realized by on-the-fly operator evaluation using numerical quadrature. The efficiency of matrix-free finite element

operator evaluation as compared to matrix-based approaches essentially depends on the polynomial degree of the shape functions and the type of elements. For quadrilateral and hexahedral element shapes, the tensor-product structure of the shape functions and of the quadrature rule can be exploited by a technique called sum-factorization, which leads to improved computational complexity in terms of operation counts and which is crucial in achieving efficient matrix-free methods.

We present here the notation and mathematical formulation of matrix-free method for a general mixed-elasticity problem $\boldsymbol{R}(\boldsymbol{u},p) = [\boldsymbol{R}_u(\boldsymbol{u},p) \quad \boldsymbol{R}_p(\boldsymbol{u},p)]^T = \boldsymbol{0}$: Find $(\boldsymbol{u},p) \in \mathbb{V} \times \mathbb{Q}$, such that

$$\begin{aligned} \langle \boldsymbol{v}, \boldsymbol{R}_u(\boldsymbol{u},p)\rangle &= \int_{\Omega_0} \boldsymbol{v}\cdot\boldsymbol{f}_0 + \nabla\boldsymbol{v}\colon\boldsymbol{f}_1\, dV - \int_{\partial\Omega_0^N} \boldsymbol{v}\cdot\bar{\boldsymbol{t}}\, dS = 0, \quad \forall \boldsymbol{v}\in\mathbb{V}_0, \\ \langle q, \boldsymbol{R}_p(\boldsymbol{u},p)\rangle &= \int_{\Omega_0} q\cdot\boldsymbol{g}_0 + \nabla q\colon\boldsymbol{g}_1\, dV = 0, \quad \forall q\in\mathbb{Q}, \end{aligned} \tag{3.78}$$

where the operators $\boldsymbol{f}_0, \boldsymbol{f}_1, \boldsymbol{g}_0, \boldsymbol{g}_1$ contain all possible sources in the problem i.e., $\boldsymbol{u}, \nabla\boldsymbol{u}, p, \nabla p$. It should be noted that the gradient in the (3.78) depends on the configuration system and it could be with respect to initial configuration $\boldsymbol{X}$ i.e., $\nabla_X\boldsymbol{v}$ (Lagrangian approach) or current configuration $\boldsymbol{x}$ i.e., $\nabla_x\boldsymbol{v}$ (Eulerian approach, see section 3.4).

To solve (3.78) with Newton Method we may state the Jacobian form for a constant $\bar{\boldsymbol{t}}$ as: Find $(\mathrm{d}\boldsymbol{u}, \mathrm{d}p) \in \mathbb{V}\times\mathbb{Q}$ such that for all $(\boldsymbol{v},q)\in\mathbb{V}_0\times\mathbb{Q}$

$$\begin{aligned} \langle \boldsymbol{v}, \boldsymbol{J}_u(\boldsymbol{u},p)\,\mathrm{d}\boldsymbol{U}\rangle &= \int_{\Omega_0} \boldsymbol{v}\cdot\mathrm{d}\boldsymbol{f}_0 + \nabla\boldsymbol{v}\colon\mathrm{d}\boldsymbol{f}_1\, dV, \quad \forall \boldsymbol{v}\in\mathbb{V}_0, \\ \langle q, \boldsymbol{J}_p(\boldsymbol{u},p)\,\mathrm{d}\boldsymbol{U}\rangle &= \int_{\Omega_0} q\cdot\mathrm{d}\boldsymbol{g}_0 + \nabla q\colon\mathrm{d}\boldsymbol{g}_1\, dV, \quad \forall q\in\mathbb{Q}, \end{aligned} \tag{3.79}$$

where $\mathrm{d}\boldsymbol{U} = [\mathrm{d}\boldsymbol{u} \quad \mathrm{d}p]^T$ and the linearization of operators $\boldsymbol{f}_0, \boldsymbol{f}_1, \boldsymbol{g}_0, \boldsymbol{g}_1$ are

$$\begin{aligned} \mathrm{d}\boldsymbol{f}_i &= \frac{\partial\boldsymbol{f}_i}{\partial\boldsymbol{u}}\,\mathrm{d}\boldsymbol{u} + \frac{\partial\boldsymbol{f}_i}{\partial\nabla\boldsymbol{u}}\nabla\,\mathrm{d}\boldsymbol{u} + \frac{\partial\boldsymbol{f}_i}{\partial p}\,\mathrm{d}p + \frac{\partial\boldsymbol{f}_i}{\partial\nabla p}\nabla\,\mathrm{d}p, \quad i=0,1 \\ \mathrm{d}\boldsymbol{g}_i &= \frac{\partial\boldsymbol{g}_i}{\partial\boldsymbol{u}}\,\mathrm{d}\boldsymbol{u} + \frac{\partial\boldsymbol{g}_i}{\partial\nabla\boldsymbol{u}}\nabla\,\mathrm{d}\boldsymbol{u} + \frac{\partial\boldsymbol{g}_i}{\partial p}\,\mathrm{d}p + \frac{\partial\boldsymbol{g}_i}{\partial\nabla p}\nabla\,\mathrm{d}p, \quad i=0,1 \end{aligned} \tag{3.80}$$

Compare with governing equations derived in pervious sections for linear and large deformation, it is easy to verify that

- For mixed linear described in (3.12) or (3.46)

$$\boldsymbol{f}_0 = -\rho \boldsymbol{g}, \quad \boldsymbol{f}_1 = 2\mu\, \boldsymbol{\varepsilon}_{\text{dev}} + (k_p \operatorname{trace} \boldsymbol{\varepsilon} - p)\, \boldsymbol{I}, \quad \boldsymbol{g}_0 = -\operatorname{trace} \boldsymbol{\varepsilon} - \frac{p}{k - k_p}, \quad \boldsymbol{g}_1 = \boldsymbol{0}$$

$$\mathrm{d}\boldsymbol{f}_1 = \boldsymbol{f}_1(\mathrm{d}\boldsymbol{u}, \mathrm{d}p), \quad \mathrm{d}\boldsymbol{g}_0 = \boldsymbol{g}_0(\mathrm{d}\boldsymbol{u}, \mathrm{d}p)$$

- For mixed hyperelastic in initial configuration described in (3.28) and (3.33)

$$\boldsymbol{f}_0 = -\rho_0 \boldsymbol{g}, \quad \boldsymbol{f}_1 = \boldsymbol{F}\boldsymbol{S}, \quad \boldsymbol{g}_0 = LJ, \quad \boldsymbol{g}_1 = \boldsymbol{0}$$

$$\mathrm{d}\boldsymbol{f}_1 = \mathrm{d}\boldsymbol{F}\boldsymbol{S} + \boldsymbol{F}\,\mathrm{d}\boldsymbol{S}, \quad \mathrm{d}\boldsymbol{g}_0 = \mathrm{d}LJ + L\,\mathrm{d}J$$

- For mixed hyperelastic with variational approach in initial configuration given in (3.51) and (3.52)

$$\boldsymbol{f}_0 = -\rho_0 \boldsymbol{g}, \quad \boldsymbol{f}_1 = \boldsymbol{F}\boldsymbol{S}, \quad \boldsymbol{g}_0 = -\hat{U}(J) - \frac{p}{k - k_p}, \quad \boldsymbol{g}_1 = \boldsymbol{0}$$

$$\mathrm{d}\boldsymbol{f}_1 = \mathrm{d}\boldsymbol{F}\boldsymbol{S} + \boldsymbol{F}\,\mathrm{d}\boldsymbol{S}, \quad \mathrm{d}\boldsymbol{g}_0 = -\,\mathrm{d}\hat{U}(J) - \frac{\mathrm{d}p}{k - k_p}$$

- For mixed hyperelastic in current configuration derived in (3.57) and (3.61)

$$\boldsymbol{f}_0 = -\rho_0 \boldsymbol{g}, \quad \boldsymbol{f}_1 = \boldsymbol{\tau}, \quad \boldsymbol{g}_0 = LJ, \quad \boldsymbol{g}_1 = \boldsymbol{0}$$

$$\mathrm{d}\boldsymbol{f}_1 = \mathrm{d}\boldsymbol{\tau} - \boldsymbol{\tau}\left(\nabla_x\, \mathrm{d}\boldsymbol{u}\right)^T, \quad \mathrm{d}\boldsymbol{g}_0 = \mathrm{d}LJ + L\,\mathrm{d}J$$

The corresponding discrete form of (3.78) without traction can be written by

$$\begin{aligned}
&\sum_e (\mathcal{E}_u^e)^T \left[\left(\boldsymbol{B}_u^e\right)^T \mathbf{W}^e \Lambda\left(\boldsymbol{f}_0\left(\boldsymbol{u}^e, \nabla \boldsymbol{u}^e\right)\right) + \sum_{i=1}^{\dim} \left(\boldsymbol{D}_u^e\right)_i^T \mathbf{W}^e \Lambda\left(\boldsymbol{f}_1\left(\boldsymbol{u}^e, \nabla \boldsymbol{u}^e\right)\right) \right] = 0 \\
&\sum_e (\mathcal{E}_p^e)^T \left[\left(\boldsymbol{B}_p^e\right)^T \mathbf{W}^e \Lambda\left(\boldsymbol{g}_0\left(\boldsymbol{p}^e, \nabla \boldsymbol{p}^e\right)\right) + \sum_{i=1}^{\dim} \left(\boldsymbol{D}_p^e\right)_i^T \mathbf{W}^e \Lambda\left(\boldsymbol{g}_1\left(\boldsymbol{p}^e, \nabla \boldsymbol{p}^e\right)\right) \right] = 0
\end{aligned} \tag{3.81}$$

where $\mathcal{E}_u^e, \mathcal{E}_p^e$ are the element e restriction operators for displacement and pressure fields that separates DoFs based on the elements they belong to, and Λ represents pointwise function evaluation. The diagonal weighting $\boldsymbol{W}^e = \det\left(\nabla_\xi \boldsymbol{X}\right) \lambda\left(\hat{W} \otimes \hat{W} \otimes \hat{W}\right)$ are quadrature weights mapped to the physical element and $\boldsymbol{\xi}$ is the coordinate of the reference element $\boldsymbol{\xi} \in [-1, 1]^{\dim}$. Both $\boldsymbol{f}_i$ and $\boldsymbol{g}_i$, $i = 0, 1$ come from the constitutive law and we compute them in quadrature point as follow. By

applying element restrictions $\mathcal{E}_u^e, \mathcal{E}_p^e$ on total assembled vector $\boldsymbol{U} = [\boldsymbol{u} \;\; p]^T$, we can get the nodal values of displacement and pressure for element e, i.e., $\boldsymbol{u}^e = \mathcal{E}_u^e \boldsymbol{U}$, $\boldsymbol{p}^e = \mathcal{E}_p^e \boldsymbol{U}$. Then map them to the quadrature points by applying interpolation basis function as $\widehat{\boldsymbol{u}^e} = \boldsymbol{B}_u^e \boldsymbol{u}^e$ and $\widehat{\boldsymbol{p}^e} = \boldsymbol{B}_p^e \boldsymbol{p}^e$, where $\widehat{\ }$symbol indicates that vectors $\widehat{\boldsymbol{u}^e}$ and $\widehat{\boldsymbol{p}^e}$ live in the quadrature space. To compute the gradient, first we apply derivative of basis function so we have gradient of nodal value in reference element $\boldsymbol{\xi} \in [-1, 1]^{\dim}$, i.e., $\widehat{\nabla_\xi \boldsymbol{u}^e} = [(\boldsymbol{D}_u^e)_d \boldsymbol{u}^e]_{d=1}^{\dim}$. The nodal basis and its derivative are constructed by

$$\begin{aligned} \boldsymbol{B} &= B \otimes B \otimes B, & \boldsymbol{D}_1 &= D \otimes B \otimes B, \\ \boldsymbol{D}_2 &= B \otimes D \otimes B, & \boldsymbol{D}_3 &= B \otimes B \otimes D, \end{aligned} \tag{3.82}$$

from one dimensional tabulations B and D of shape functions and their derivatives at quadrature points. Then we compute the physical gradient by

$$\widehat{\nabla_X \boldsymbol{u}^e} = J_e^{-1} \widehat{\nabla_\xi \boldsymbol{u}^e}$$

where $J_e^{-1} = \nabla_X \boldsymbol{\xi}$ is inverse of the jacobian i.e., $J_e = [\boldsymbol{D}_d^e \boldsymbol{X}^e]_{d=1}^{\dim}$ with the physical coordinate $\boldsymbol{X}^e$ of element e. Pulling (3.81) back to reference coordinates, we have

$$\begin{aligned} \boldsymbol{R}_u &= \sum_e (\mathcal{E}_u^e)^T \begin{bmatrix} \boldsymbol{B}_u \\ \boldsymbol{D}_u \end{bmatrix}^T \boldsymbol{W}^e \Lambda \begin{bmatrix} \left(\hat{\boldsymbol{f}}_0 \left(\boldsymbol{u}^e, \widehat{\nabla_\xi \boldsymbol{u}}^e\right)\right) \\ \left(\hat{\boldsymbol{f}}_1 \left(\boldsymbol{u}^e, \widehat{\nabla_\xi \boldsymbol{u}}^e\right)\right) \end{bmatrix} \\ \boldsymbol{R}_p &= \sum_e (\mathcal{E}_p^e)^T \begin{bmatrix} \boldsymbol{B}_p \\ \boldsymbol{D}_p \end{bmatrix}^T \boldsymbol{W}^e \Lambda \begin{bmatrix} \left(\hat{\boldsymbol{g}}_0 \left(\boldsymbol{p}^e, \widehat{\nabla_\xi \boldsymbol{p}}^e\right)\right) \\ \left(\hat{\boldsymbol{g}}_1 \left(\boldsymbol{p}^e, \widehat{\nabla_\xi \boldsymbol{p}}^e\right)\right) \end{bmatrix} \end{aligned} \tag{3.83}$$

where for instance $\hat{\boldsymbol{f}}_0 \left(\boldsymbol{u}^e, \widehat{\nabla_\xi \boldsymbol{u}^e}\right) = \boldsymbol{f}_0 \left(\boldsymbol{u}^e, J_e^{-1} \widehat{\nabla_\xi \boldsymbol{u}^e}\right)$ and $\hat{\boldsymbol{f}}_1 \left(\boldsymbol{u}^e, \widehat{\nabla_\xi \boldsymbol{u}^e}\right) = J_e^{-1} \boldsymbol{f}_1 \left(\boldsymbol{u}^e, J_e^{-1} \widehat{\nabla_\xi \boldsymbol{u}^e}\right)$.

While this interface brings the isoparametric mapping into quadrature functions, the work outside these quadrature routines shares the same data and can be batched over elements, leading to improved vectorization and data reuse.

In this formulation, the element restriction operators and basis operators can represent different element geometries and different degree polynomial bases, providing a flexible description

for arbitrary meshes. It should be noted that we used the notation $\boldsymbol{B}_u^e$ and $\boldsymbol{B}_p^e$ to indicate that the polynomial degree can be different for displacement and pressure fields.

The Jacobian action can be computed [24] similar to the residual (3.83)

$$\begin{aligned}
\boldsymbol{J}_u \,\mathrm{d}\boldsymbol{U} &= \sum_e \left(\mathcal{E}_u^e\right)^T \begin{bmatrix} \boldsymbol{B}_u^e \\ \boldsymbol{D}_u^e \end{bmatrix}^T \boldsymbol{W}^e \Lambda \begin{bmatrix} \hat{\boldsymbol{f}}_{0,0} & \hat{\boldsymbol{f}}_{0,1} \\ \hat{\boldsymbol{f}}_{1,0} & \hat{\boldsymbol{f}}_{1,1} \end{bmatrix} \begin{bmatrix} \boldsymbol{B}_u^e \\ \boldsymbol{D}_u^e \end{bmatrix} \mathcal{E}_u^e \,\mathrm{d}\boldsymbol{U} \\
\boldsymbol{J}_p \,\mathrm{d}\boldsymbol{U} &= \sum_e \left(\mathcal{E}_p^e\right)^T \begin{bmatrix} \boldsymbol{B}_p^e \\ \boldsymbol{D}_p^e \end{bmatrix}^T \boldsymbol{W}^e \Lambda \begin{bmatrix} \hat{\boldsymbol{g}}_{0,0} & \hat{\boldsymbol{g}}_{0,1} \\ \hat{\boldsymbol{g}}_{1,0} & \hat{\boldsymbol{g}}_{1,1} \end{bmatrix} \begin{bmatrix} \boldsymbol{B}_p^e \\ \boldsymbol{D}_p^e \end{bmatrix} \mathcal{E}_p^e \,\mathrm{d}\boldsymbol{U}
\end{aligned} \tag{3.84}$$

where the $\hat{\boldsymbol{f}}_{i,0}, \hat{\boldsymbol{f}}_{i,1}$ and $\hat{\boldsymbol{g}}_{i,0}, \hat{\boldsymbol{g}}_{i,1}$ are pull back of derivative of $\boldsymbol{f}_i$ and $\boldsymbol{g}_i$ with respect to the solution fields given in (3.80). The general mixed formulation resulting from discretization at iteration m with given initial guess $\boldsymbol{U}^0$ has the form

$$\begin{aligned}
\boldsymbol{J}(\boldsymbol{U}^m)\,\mathrm{d}\boldsymbol{U}^m &= -\boldsymbol{R}(\boldsymbol{U}^m) \\
\boldsymbol{U}^{m+1} &= \boldsymbol{U}^m + \mathrm{d}\boldsymbol{U}^m
\end{aligned} \tag{3.85}$$

where $\boldsymbol{R}(\boldsymbol{U}^m)$ and Jacobian $\boldsymbol{J}$ are discrete form of system (3.41) (or (3.55) and (3.72) if we use perturbed Lagrange-multiplier method and formulation in current configuration) needs to be solved at each Newton's iteration.

Systems of equations with the form (3.83) and (3.84) admit a natural implementation via libCEED [25], which provides fast algebra for element-based discretizations on CPUs and GPUs. Figure 3.2 illustrates the action of an arbitrary finite element operator,

$$\mathbf{A} = \mathcal{P}^T \mathcal{E}^T \mathbf{B}^T \mathbf{D} \mathbf{B} \mathcal{E} \mathcal{P}, \tag{3.86}$$

where $\mathcal{P}$ represents the parallel communication portion of the element restriction operator, $\mathcal{E}$ represents the local portion of the element restriction operator, $\mathbf{B}$ represents the basis action kernels that provide solution values and derivatives at the quadrature points given by $\boldsymbol{B}_u, \boldsymbol{B}_p$ and $\boldsymbol{D}_u, \boldsymbol{D}_p$, and $\mathbf{D}$ (which may be linear or nonlinear) represents the pointwise representation of the weak form,

given by $\hat{\boldsymbol{f}}_i, \hat{\boldsymbol{g}}_i$ and $\hat{\boldsymbol{f}}_{i,j}, \hat{\boldsymbol{g}}_{i,j}$ as well as the element quadrature weights $\mathbf{W}$ and geometric factors $\nabla_\xi \boldsymbol{X}$.

$$A = \mathcal{P}^T \mathcal{E}^T B^T D B \mathcal{E} \mathcal{P}$$

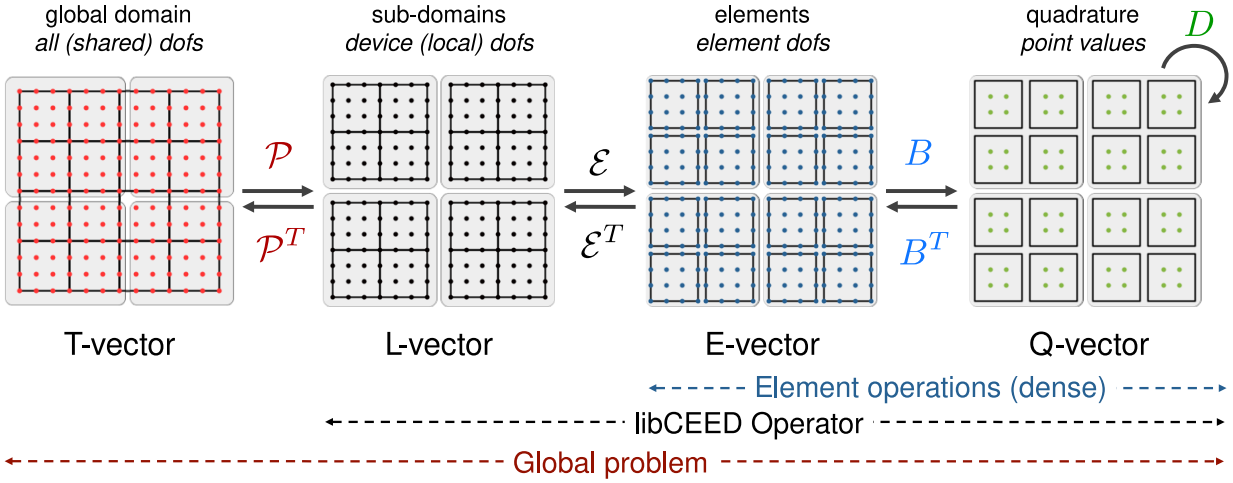


Figure 3.2: libCEED composes local (L-vector to L-vector) operations from element restriction $\mathcal{E}$, basis $\boldsymbol{B}$, and quadrature-point functions $\boldsymbol{D}$. A T-vector represents the non-overlapping parallel partition of DoFs, as needed by nonlinear and linear algebraic solvers. The L-vector is localized per device (e.g., MPI rank or GPU context) with any ghost values replicated into each part. The E-vector (restricted to elements) and Q-vector (evaluated to quadrature points) exist only conceptually in our optimized implementation, since restriction $\mathcal{E}$, basis $\mathcal{B}$, and user-provided quadrature function $\mathbf{D}$ are fused into one kernel.

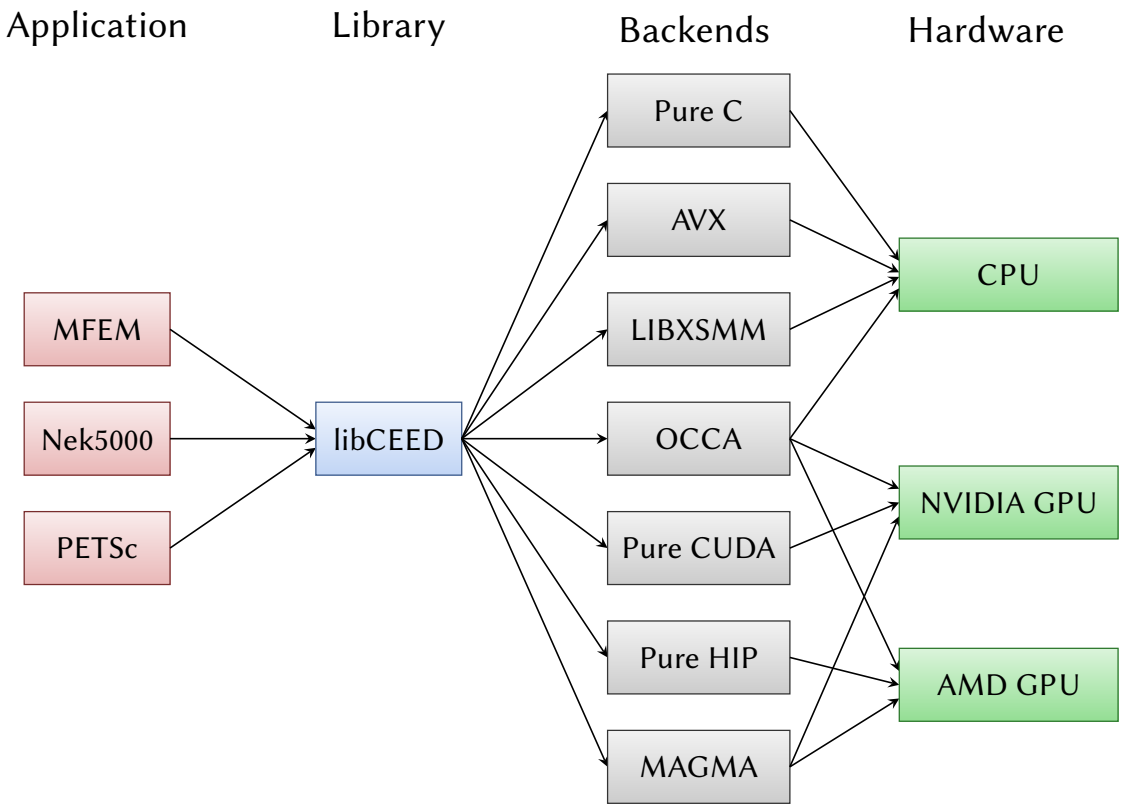


Figure 3.3: libCEED Backends.

3.6 Preconditioner

Linear system of the form

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b} \tag{3.87}$$

could typically be solved through direct or iterative methods if a solution exists. For large scale problems, the iterative methods known as Krylov subspace methods are more efficient. In theory, the Krylov subspace methods are supposed to converge in a finite number of iterations. The Conjugate Gradient method (CG) and Generalized Minimal Residual method (GMRES) are well-known Krylov subspace linear solvers. If matrix $\boldsymbol{A}$ is symmetric positive definite (SPD), the CG method is preferred, while GMRES can be considered for a more general non-symmetric matrix $\boldsymbol{A}$. Krylov iteration methods are not necessarily fast on their own, and the iterations count depends on condition number of the operator. Incompressibility in elasticity problems typically produce

difficult matrices to solve. That means the numerical properties of the matrix $\boldsymbol{A}$ can have an effect on how many iterations the Krylov subspace method will take to reach to a solution. In addition, as the size of the system of equations in (3.87) increases, the number of iterations to reach a solution by a Krylov subspace solver increases. To accelerate the solve time and bound the number of iterations for a Krylov subspace method, regardless of the system size, preconditioning techniques are employed. Preconditioners transform (3.87) to a new system of equations with smaller condition number and therefore reduce total iteration count and total solve time. This transformation takes place via a non-singular matrix, $\boldsymbol{P}^{-1}$, known as a preconditioner

$$\boldsymbol{P}^{-1}\boldsymbol{A}\boldsymbol{x} = \boldsymbol{P}^{-1}\boldsymbol{b} \tag{3.88}$$

For the preconditioner to be effective we need

- $\boldsymbol{P}^{-1}\boldsymbol{z}$ is easy to compute for any vector $\boldsymbol{z}$.
- The condition number of the preconditioned problem is smaller than the original problem.

There are numerous techniques for preconditioning which can be found in [78, 16, 13]. For elliptic PDEs, multigrid methods provide an efficient preconditioning framework for obtaining uniform convergence rates with respect to resolution and model extent. Multigrid methods can be categorized into $h-$multigrid, $p-$multigrid, and algebraic multigrid (AMG) techniques. Geometric or $h-$multigrid methods rely on a hierarchy of meshes. In many cases, such as unstructured meshes used in finite element methods, $h-$multigrid is hard to implement. In contrast to geometric multigrid methods, AMG techniques extract all information from the assembled system matrix without requiring any mesh information. To avoid assembling high-order matrix-free operator, we make use of $p-$multigrid which coarsens the operator by reducing the order of the basis functions and AMG solver is needed at the coarsest level (linear element). Algorithm 1 shows the preconditioned conjugate gradient [78, 16, 36] for solving (3.87) which require the action of $\boldsymbol{A}$ applied to a vector, so that an explicit assembly of the matrix $\boldsymbol{A}$ is avoided by the use of matrix-free operator evaluation. The operator only needs to be assembled on the coarsest multigrid level for the AMG

coarse-level solver in line 3 of Algorithm 2. In the preconditioning step of the conjugate gradient solver (preconditioner $\boldsymbol{P}$), the operator is inverted approximately by performing one multigrid V-cycle according to Algorithm 2 with initial solution $\boldsymbol{x}^{(l)} = \boldsymbol{0}$, where l denotes the finest level. Pre- and post-smoothing are done in lines 6 and 11, respectively, with 2nd order Chebyshev/Jacobi iteration in the range of $[0.1\lambda_{\max}, 1.1\lambda_{\max}]$, where $\lambda_{\max}$ is the eigenvalue estimate of $(\text{diag}\boldsymbol{A})^{-1}\boldsymbol{A}$. Note that the prolongation operator in line 10 of Algorithm 2 is given by

$$\boldsymbol{P}^{(l,l-1)} = \sum_e \left(\mathcal{E}_f^e\right)^T \lambda\left(m_f^{-1}\right) \boldsymbol{B}_{\text{ctof}}^e \mathcal{E}_c^e, \tag{3.89}$$

where, $\boldsymbol{B}_{\text{ctof}}^e$ is the interpolation kernel from the lower order to high order finite element, defined by (3.82), and $m_f = \mathcal{E}_f \mathcal{E}_f^T \mathcal{E}_f$ is a pointwise scaling factor for the multiplicity of nodes shared between elements on the fine grid.

Algorithm 1 Preconditioned conjugate gradient algorithm for solving $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$

Input: Given tolerances ε_{rel}, ε_{abs}

function PCG($\boldsymbol{A}, \boldsymbol{x}, \boldsymbol{b}$)

  $\boldsymbol{r} = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}$

  $\|\boldsymbol{r}_0\| = \|\boldsymbol{r}\|$

  $\boldsymbol{z} = \boldsymbol{P}^{-1}\boldsymbol{r}$ ▷ e.g., MultigridVCycle($L, \boldsymbol{A}, \boldsymbol{0}, \boldsymbol{r}$)

  $\boldsymbol{p} = \boldsymbol{z}$

  $\delta = \boldsymbol{r}^T\boldsymbol{z}$

  while $\|\boldsymbol{r}\|/\|\boldsymbol{r}_0\| > \varepsilon_{\text{rel}}$ **and** $\|\boldsymbol{r}\| > \varepsilon_{\text{abs}}$ **do**

    $\boldsymbol{z} = \boldsymbol{A}\boldsymbol{p}$

    $\alpha = \delta/(\boldsymbol{p}^T\boldsymbol{z})$

    $\boldsymbol{x} \leftarrow \boldsymbol{x} + \alpha\boldsymbol{p}$

    $\boldsymbol{r} \leftarrow \boldsymbol{r} - \alpha\boldsymbol{z}$

    $\|\boldsymbol{r}\|$

    $\boldsymbol{z} = \boldsymbol{P}^{-1}\boldsymbol{r}$ ▷ e.g., MultigridVCycle($L, \boldsymbol{A}, \boldsymbol{0}, \boldsymbol{r}$)

    $\delta_{\text{old}} = \delta$

    $\delta = \boldsymbol{r}^T\boldsymbol{z}$

    $\beta = \delta/\delta_{\text{old}}$

    $\boldsymbol{p} \leftarrow \boldsymbol{z} + \beta\boldsymbol{p}$

  end while

end function

Algorithm 2 Multigrid V-cycle

function MultigridVCycle($l, \boldsymbol{A}^{(l)}, \boldsymbol{x}^{(l)}, \boldsymbol{b}^{(l)}$)
  if $l = 0$ **then**
    $\boldsymbol{x}^{(0)} \leftarrow$ CoarseLevelSolver($\boldsymbol{A}^{(0)}, \boldsymbol{x}^{(0)}, \boldsymbol{b}^{(0)}$) ▷ e.g., AMG
    return $\boldsymbol{x}^{(0)}$
  else
    $\boldsymbol{x}^{(l)} \leftarrow$ Smooth($\boldsymbol{A}^{(l)}, \boldsymbol{x}^{(l)}, \boldsymbol{b}^{(l)}, \lambda_{\max}$) ▷ pre-smoothing
    $\boldsymbol{r}^{(l)} = \boldsymbol{b}^{(l)} - \boldsymbol{A}^{(l)}\boldsymbol{x}^{(l)}$
    $\boldsymbol{r}^{(l-1)} = \boldsymbol{R}^{(l-1,l)}\boldsymbol{r}^{(l)}$ ▷ restriction
    $\boldsymbol{x}^{(l-1)} \leftarrow$ MultigridVCycle($l-1, \boldsymbol{A}^{(l-1)}, \boldsymbol{0}, \boldsymbol{r}^{(l-1)}$)
    $\boldsymbol{x}^{(l)} \leftarrow \boldsymbol{x}^{(l)} + \boldsymbol{P}^{(l,l-1)}\boldsymbol{x}^{(l-1)}$ ▷ prolongation
    $\boldsymbol{x}^{(l)} \leftarrow$ Smooth($\boldsymbol{A}^{(l)}, \boldsymbol{x}^{(l)}, \boldsymbol{b}^{(l)}, \lambda_{\max}$) ▷ post-smoothing
    return $\boldsymbol{x}^{(l)}$
  end if
end function

3.7 Block preconditioner

To design the preconditioner for the block structure system (3.41), we start by rewriting the factorization of the jacobian matrix by [16, 69, 50]

$$
\begin{aligned}
\boldsymbol{J} = \begin{bmatrix} \mathcal{A} & \mathcal{B} \\ \mathcal{C} & \mathcal{D} \end{bmatrix} &= \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \mathcal{C}\mathcal{A}^{-1} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \mathcal{A} & \boldsymbol{0} \\ \boldsymbol{0} & \mathcal{S} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & \mathcal{A}^{-1}\mathcal{B} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \mathcal{C}\mathcal{A}^{-1} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & -\boldsymbol{B}\boldsymbol{S}^{-1} \\ \boldsymbol{0} & -\boldsymbol{I} \end{bmatrix} \begin{bmatrix} \mathcal{A} & \boldsymbol{0} \\ \boldsymbol{0} & -\mathcal{S} \end{bmatrix} \\
&= \begin{bmatrix} \mathcal{A} & \boldsymbol{0} \\ \mathcal{C} & \mathcal{S} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & \mathcal{A}^{-1}\mathcal{B} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0} \\ \mathcal{C}\mathcal{A}^{-1} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \mathcal{A} & \mathcal{B} \\ \boldsymbol{0} & \mathcal{S} \end{bmatrix}
\end{aligned}
\tag{3.90}
$$

where $\mathcal{S} = \mathcal{D} - \mathcal{C}\mathcal{A}^{-1}\mathcal{B}$ is called the Schur complement. The simplest block preconditioner is obtained by only considering the diagonal blocks $\mathcal{A}$ and $\mathcal{S}$ as [69, 50]

$$\boldsymbol{P} = \begin{bmatrix} \mathcal{A} & \mathbf{0} \\ \mathbf{0} & -\mathcal{S} \end{bmatrix} \longrightarrow \boldsymbol{P}^{-1} = \begin{bmatrix} \mathcal{A}^{-1} & \mathbf{0} \\ \mathbf{0} & -\mathcal{S}^{-1} \end{bmatrix} \tag{3.91}$$

which can applied as a preconditioner in Krylov subspace iterative method such as MINRES for a symmetric problem. The preconditioned operator $\boldsymbol{J}\boldsymbol{P}^{-1}$ has minimal polynomial degree 3 so that MINRES converges in three iterations [69]. From the given factorization in (3.90), the block triangular preconditioner can also be taken into account as

$$\boldsymbol{P} = \begin{bmatrix} \mathcal{A} & \mathcal{B} \\ \mathbf{0} & -\mathcal{S} \end{bmatrix} \longrightarrow \boldsymbol{J}\boldsymbol{P}^{-1} = \begin{bmatrix} \boldsymbol{I} & \mathbf{0} \\ \mathcal{C}\mathcal{A}^{-1} & -\boldsymbol{I} \end{bmatrix} \tag{3.92}$$

which the preconditioned system has the minimal polynomial $(\lambda - 1)(\lambda + 1)$ [50] thus converges faster compared to block diagonal preconditioner. However, if we write the inverse of the triangular preconditioner

$$\boldsymbol{P}^{-1} = \begin{bmatrix} \mathcal{A}^{-1} & \mathcal{A}^{-1}\mathcal{B}\mathcal{S}^{-1} \\ \mathbf{0} & -\mathcal{S}^{-1} \end{bmatrix} = \begin{bmatrix} \mathcal{A}^{-1} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & \mathcal{B} \\ \mathbf{0} & -\boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{I} & 0 \\ \mathbf{0} & \mathcal{S}^{-1} \end{bmatrix} \tag{3.93}$$

we need to apply extra operator $\mathcal{B}$ to the block system and increase the cost of computation.

In this study we consider the upper triangular preconditioner of the form

$$\begin{bmatrix} \hat{\mathcal{A}} & \mathcal{B} \\ \mathbf{0} & \hat{\mathcal{D}} \end{bmatrix}^{-1} \tag{3.94}$$

where the blocks $\hat{\mathcal{A}}$ and $\hat{\mathcal{D}}$ are related to the original blocks and for instance, in linear elasticity, are defined by

$$\begin{aligned} \hat{\mathcal{A}} &\sim \int_\Omega \nabla \boldsymbol{v} : (2\mu\, \boldsymbol{\varepsilon}_{\text{dev}} + k_{pc}\, \text{trace}(\boldsymbol{\varepsilon})\boldsymbol{I})\; dv, \\ \hat{\mathcal{D}} &\sim -\int_\Omega q \left(\frac{1}{k - k_{pc}} + \frac{1}{\mu} \right) \text{d}p\, dv, \end{aligned} \tag{3.95}$$

where k_{pc} is bulk modulus defined in terms of ν_{pc} with $-1 \leq \nu_{pc} < \nu$ to indicate that $\hat{\mathcal{A}}$ could be different from original displacement block $\mathcal{A}$ given in terms of k_p in the first equation of (3.13).

Note that this approach may be considered an augmented Lagrangian technique parametrized by ν_{pc} that is always consistent for any physical Poisson's ratio ν. The scaling in (3.95) is intended to deliver maximum eigenvalues of the preconditioned operator close to 1, similar to multigrid applied to the displacement block $\hat{\mathcal{A}}$. Note that in finite strain, $\hat{\mathcal{A}}$ and $\hat{\mathcal{D}}$ are defined similar to (3.42) in terms of k_{pc} i.e.,

$$\begin{aligned} \hat{\mathcal{A}} &\sim \int_{\Omega_0} \nabla_X \boldsymbol{v} : (\mathrm{d}\boldsymbol{F}\boldsymbol{S} + \boldsymbol{F}\,\mathrm{d}\boldsymbol{S}_{\mathrm{iso}} + \boldsymbol{F}\,\mathrm{d}\boldsymbol{S}^{u}_{\mathrm{vol}}(k_{pc}))\; dV, \\ \hat{\mathcal{D}} &\sim -\int_{\Omega_0} q \left(\frac{1}{k - k_{pc}} + \frac{1}{\mu} \right) J\,\mathrm{d}p\, dV. \end{aligned} \tag{3.96}$$

To approximate the displacement and pressure blocks $\hat{\mathcal{A}}^{-1}$, $\hat{\mathcal{D}}^{-1}$, Ratel uses p-multigrid and variable size point block Jacobi preconditioners. In the next chapter, we present numerical experiment for different loading in small and finite strain to show the performance and accuracy of the proposed preconditioner.

Chapter 4

Numerical experiments

In this chapter we present different numerical experiments for incompressible material in small and finite strain regimes.

4.1 Small strain

Before we present the results for mixed formulation, we investigate the error estimate formula (3.5) for linear elasticity with exact solution

$$u_x = A_0 \sin(sx - x_0) \sin(sy - y_0), \quad u_y = 2\, u_x. \tag{4.1}$$

with $A_0 = 10^{-3}$, $s = 2\pi$, $x_0 = y_0 = \frac{\pi}{2}$ on unit square $[0,1]^2$ with $n_x \times n_y$ uniform mesh, where n_x, n_y are number of elements per side Figure 4.1. The material properties is $\lambda = 3, \mu = 1, \nu = 0.375$ and full Dirichlet boundary condition is applied.

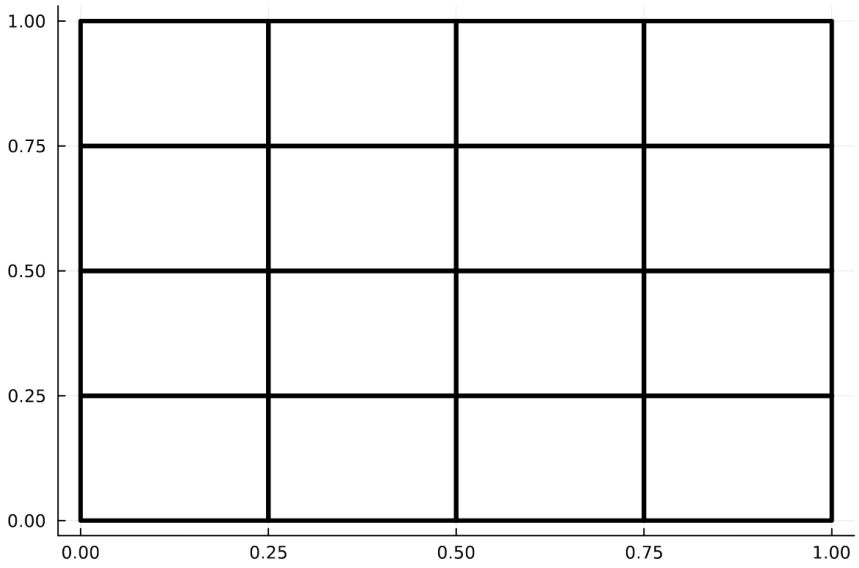


Figure 4.1: Example of uniform mesh on unit square with 4 elements per side, $n_x = n_y = 4$

Table 4.1 shows the L2 error and convergence order for linear elasticity with Q_2, Q_3, Q_4 elements. The convergence order is computed via

$$\log_{10}\left(\frac{\|\boldsymbol{u} - \boldsymbol{u}_{h_2}\|}{\|\boldsymbol{u} - \boldsymbol{u}_{h_1}\|}\right) \Big/ \log_{10}\left(\frac{h_2}{h_1}\right) \tag{4.2}$$

where $h = 1/n_x$, $n_x = n_y$ is the element size. As expected, from (3.5) the convergence order for Q_n is $n + 1$ and the results are in good agreement with theory.

Table 4.1: Convergence study for linear elasticity with $n_x \times n_x$ uniform mesh.

n_x	Q_2		Q_3		Q_4	
	Error $\boldsymbol{u}$	Order $\boldsymbol{u}$	Error $\boldsymbol{u}$	Order $\boldsymbol{u}$	Error $\boldsymbol{u}$	Order $\boldsymbol{u}$
4	3.541e−05		2.908e−06		2.023e−07	
8	3.917e−06	3.176	1.687e−07	4.108	6.041e−09	5.066
12	1.123e−06	3.080	3.244e−08	4.066	7.861e−10	5.030
16	4.682e−07	3.042	1.016e−08	4.037	1.857e−10	5.016

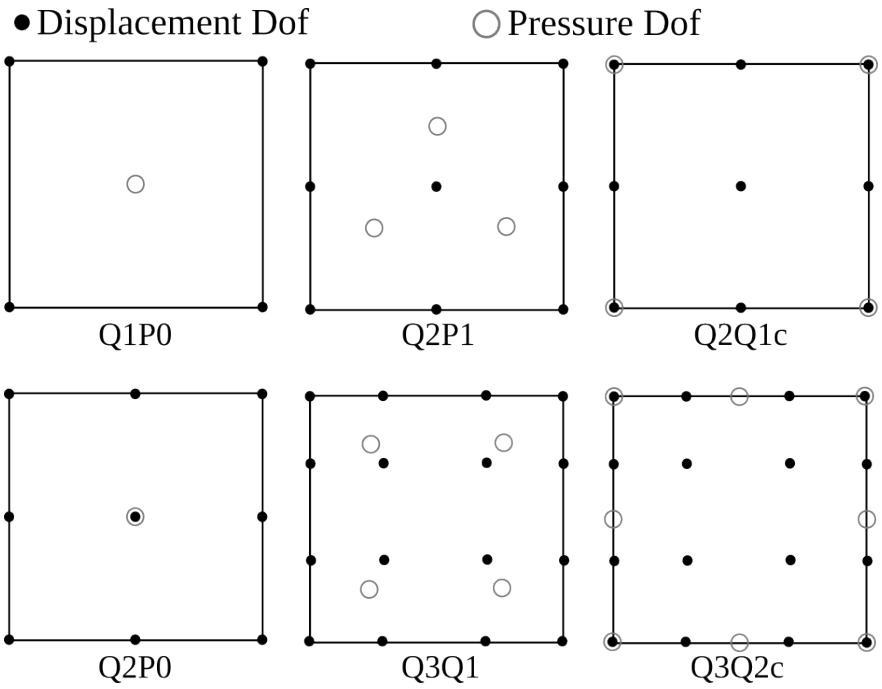


Figure 4.2: 2D mixed element examples with discontinuous and continuous (last column) pressure

For the mixed linear elasticity we used the same exact solution (4.1) for the displacement field and computed the pressure by

$$p = -(k - k_p)\nabla \cdot \boldsymbol{u}. \tag{4.3}$$

Before we compute the error, we should check the stability of the mixed elements created by combination of different polynomial orders for displacement and pressure fields. Figure 4.2 shows some mixed elements where the pressure DoF can be continuous or discontinuous. We performed the inf-sup analysis by solving the eigen problem (3.19) for various mixed elements on square $[0, 1]^2$ and stretched meshes $[0, 1] \times [0, 0.1]$ Figure 4.3. All mixed elements with continuous pressure are inf-sup stable on element with aspect ratio 1 and 10. However, for the discontinuous pressure, the inf-sup constant for widely-used Q_1P_0 and Q_nQ_{n-1} elements is decreasing under mesh refinement which make them unstable. Moreover, Q_2P_1 element which is stable on square mesh, becomes unstable on the stretched meshes. In fact, the inf-sup constant of Q_2P_1 is decreasing by factor $1/\sqrt{a}$ where a is element aspect ratio. Thus, for stretched structure one should avoid using this

element.

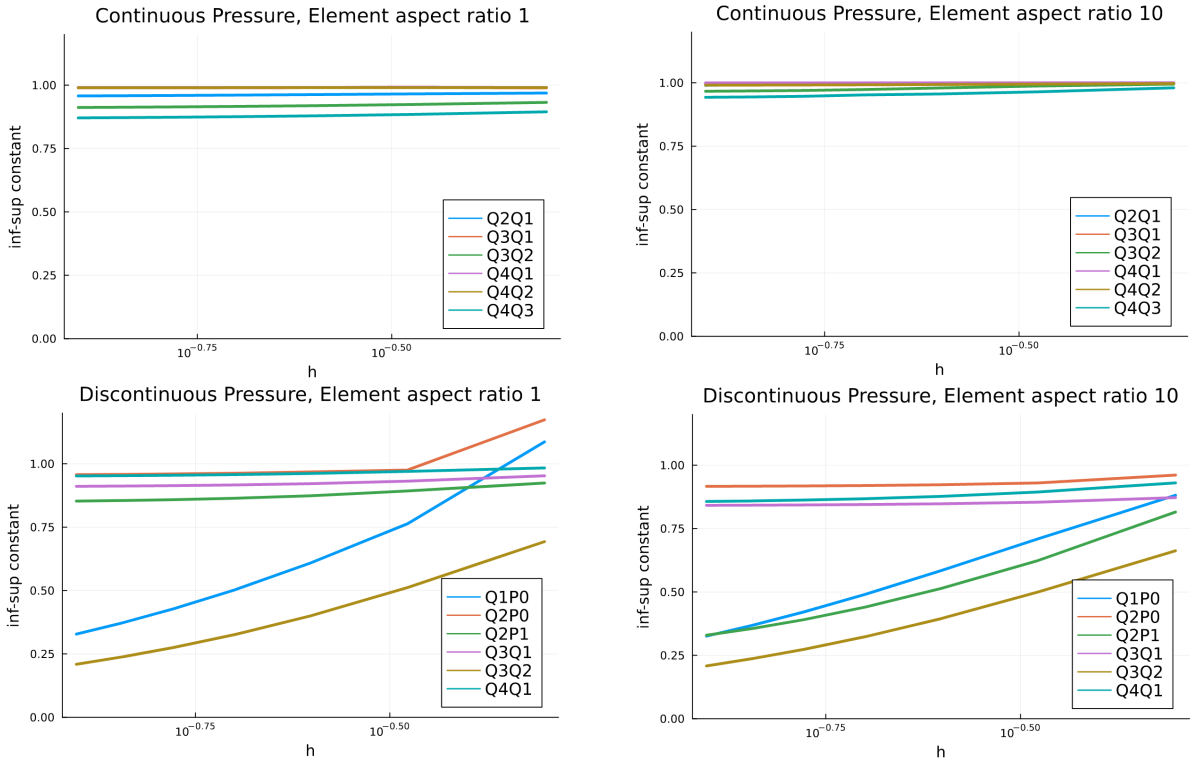


Figure 4.3: Inf-sup constant for various mixed-elements with continuous (top row) and discontinuous (bottom row) pressure on square and stretched meshes.

We investigated the L_2 error and convergence order for mixed formulation on the the uniform mesh Figure 4.1 with inf-sup stable mixed elements. Table 4.2 shows the convergence order and error for mixed element with discontinuous pressure field. It is clear that Q_2P_1 element with total DoFs/element 21 (in 2D) has the optimal convergence orders $3, 2$ and its L_2 error is close to the Q_3Q1 element (same convergence orders) with total DoFs/element 36. In general we can conclude that Q_nP_{n-1} has the order of $n+1$ for displacement and n for pressure and Q_nQ_{n-2} convergence orders for $\boldsymbol{u}, p$ are $n, n-1$, respectively.

For continuous pressure, we are allowed to have elements of type Q_nQ_{n-1} and Q_nQ_{n-2} based on inf-sup analysis Figure 4.3. However, as shown in Table 4.3, comparing Q_2Q_{1c} and Q_3Q_1,

increasing the interpolation order of displacement field doesn't gain better convergence order for displacement nor better accuracy in general. In fact, convergence orders of Q_3Q_1 is $3, 2$ which is similar to Q_2Q_{1c} but with more DoFs . Therefore, Q_nQ_{n-1} with convergence orders $n+1, n$ for $\boldsymbol{u}, p$ is a bettor choice for discretizing the mixed finite element problem. All the results in this section generated by a finite element code written in julia programming language and are reproducible [80].

Table 4.2: Convergence study for mixed linear elasticity in the incompressibility regime with **discontinuous** pressure with $n_x \times n_x$ uniform mesh.

n_x	Q_2P_0				Q_2P_1			
	Error $\boldsymbol{u}$	Order $\boldsymbol{u}$	Error p	Order p	Error $\boldsymbol{u}$	Order $\boldsymbol{u}$	Error p	Order p
4	7.331e−04		1.512e−02		1.042e−04		5.774e−03	
8	2.224e−04	1.721	8.085e−03	0.904	7.872e−06	3.727	1.544e−03	1.903
12	1.035e−04	1.886	5.456e−03	0.970	1.765e−06	3.688	6.928e−04	1.976
16	5.932e−05	1.935	4.109e−03	0.986	6.377e−07	3.538	3.907e−04	1.991
n_x	Q_3Q_1				Q_4Q_2			
	Error $\boldsymbol{u}$	Order $\boldsymbol{u}$	Error p	Order p	Error $\boldsymbol{u}$	Order $\boldsymbol{u}$	Error p	Order p
4	8.397e−05		3.228e−03		7.979e−06		4.319e−04	
8	1.156e−05	2.862	8.300e−04	1.960	5.308e−07	3.910	5.519e−05	2.968
12	3.496e−06	2.949	3.707e−04	1.988	1.064e−07	3.963	1.642e−05	2.990
16	1.487e−06	2.971	2.089e−04	1.994	3.388e−08	3.979	6.936e−06	2.995

Table 4.3: Convergence study for mixed-linear elasticity in incompressibility regime with **continuous** pressure with $n_x \times n_x$ uniform mesh.

n_x	Q_2Q_{1c}				Q_3Q_{2c}			
	Error $\boldsymbol{u}$	Order $\boldsymbol{u}$	Error p	Order p	Error $\boldsymbol{u}$	Order $\boldsymbol{u}$	Error p	Order p
4	9.467e−05		4.168e−03		6.935e−06		5.669e−04	
8	5.756e−06	4.040	8.953e−04	2.219	6.971e−07	3.314	8.775e−05	2.692
12	1.359e−06	3.559	3.839e−04	2.088	1.584e−07	3.654	2.770e−05	2.844
16	5.231e−07	3.319	2.131e−04	2.046	5.323e−08	3.792	1.201e−05	2.905
n_x	Q_3Q_{1c}				Q_4Q_{2c}			
	Error $\boldsymbol{u}$	Order $\boldsymbol{u}$	Error p	Order p	Error $\boldsymbol{u}$	Order $\boldsymbol{u}$	Error p	Order p
4	1.470e−04		4.150e−03		1.263e−05		5.657e−04	
8	1.381e−05	3.412	8.942e−04	2.214	1.159e−06	3.446	8.772e−05	2.689
12	3.803e−06	3.181	3.837e−04	2.087	2.562e−07	3.723	2.769e−05	2.844
16	1.562e−06	3.095	2.131e−04	2.045	8.510e−08	3.831	1.201e−05	2.905

To study the performance of the block preconditioner (3.94) in small strain, we begin with the effect of ν_p and ν_{pc} for the general mixed linear formulation (3.11) on the linear solve, condition number and the range of eigenvalues for preconditioned system for different boundary conditions in 3D with element per sides $n_x = n_y = n_z = 3$. We use Cholesky factorization for both the displacement $\hat{\mathcal{A}}$ and pressure $\hat{\mathcal{D}}$ blocks. Tables Table 4.4 and Table 4.5 show the results for full Dirichlet (with an MMS displacement) and clamp-traction (bending mode) boundary conditions with Young's modulus $E = 1$ and Poisson's ratio $\nu = 0.49$ for a 3D unit box with mesh size $h = 1/6$ and Q_2Q_{1c} element, where $_{.c}$ indicates that pressure is continuous. The full strain formulation (3.8) ($\nu_p = \nu_{pc} = 0$) and deviatoric form (3.9) ($\nu_p = \nu_{pc} = -1$) are indicated by bold font. The diagonal of the tables are for the case when the displacement block in Jacobian matrix (3.13) is equal to the

provided operator (3.95) i.e., $\mathcal{A} = \hat{\mathcal{A}}$. As shown in table Table 4.4 the iteration count, condition number and smallest and largest eigenvalues when $\nu_p = \nu_{pc}$ (diagonal of the table) are similar and they are deviated a little when $\nu_p \neq \nu_{pc}$.

Table 4.4: KSP iteration, condition number κ, smallest and largest eigenvalues i.e., $[\beta_{\min}, \beta_{\max}]$ for different ν_p and ν_{pc} for mixed linear elasticity with Q_2Q_{1c} element (continuous pressure), and non-homogeneous Dirichlet boundary conditions (MMS problem).

	$\nu_p = -1.0$	$\nu_p = -0.3$	$\nu_p = 0.0$	$\nu_p = 0.3$
$\nu_{pc} = -1.0$	**20** **9.39** **[0.12, 1]**	29 17.17 [0.12, 1.45]	35 10.05 [0.17, 1.82]	50 63.89 [0.09, 3.12]
$\nu_{pc} = -0.3$	29 13.21 [0.12, 0.99]	19 9.52 [0.11, 1]	29 18.18 [0.11, 1.41]	44 64.32 [0.09, 2.50]
$\nu_{pc} = 0.0$	34 5.0 [0.25, 0.95]	29 13.75 [0.12, 0.99]	**18** **9.85** **[0.11, 1]**	38 25.43 [0.14, 1.98]
$\nu_{pc} = 0.3$	44 22.55 [0.12, 0.99]	41 13.25 [0.12, 0.99]	37 11.19 [0.15, 0.97]	15 11.29 [0.09, 1]

However, when the boundary condition is changed to clamp-traction (table Table 4.5) the deviatoric formulation has a huge condition number and eigenvalue due to dilation null space. Keeping this formulation (first column), we can increase ν_{pc} in $\hat{\mathcal{A}}$ to get a maximum eigenvalue close to 1 and smaller condition number. The results for full strain formulation i.e., $\nu_p = \nu_{pc} = 0$ remains in the same range that we have for non-homogeneous Dirichlet boundary conditions. It

should be noted the smallest iteration count and condition number with largest eigenvalue 1 are obtained for the discretization with $\nu_p = \nu_{pc} = 0.3$.

Table 4.5: KSP iteration, condition number κ, smallest and largest eigenvalues i.e., $[\beta_{\min}, \beta_{\max}]$ for different ν_p and ν_{pc} for mixed linear elasticity with Q_2Q_{1c} element (continuous pressure), and clamp-traction boundary conditions (bending mode).

	$\nu_p = -1.0$	$\nu_p = -0.3$	$\nu_p = 0.0$	$\nu_p = 0.3$
$\nu_{pc} = -1.0$	**81** **980.13** **[0.42, 401.92]**	112 1282.02 [0.39, 521.73]	147 1811.23 [0.33, 675.66]	269 6361.95 [0.24, 1290.96]
$\nu_{pc} = -0.3$	43 12.57 [0.36, 1.60]	27 12.57 [0.36, 3.27]	42 15.18 [0.31, 4.73]	73 58.07 [0.21, 10.16]
$\nu_{pc} = 0.0$	57 40.42 [0.18, 0.99]	33 7.12 [0.52, 0.77]	**20** **7.34** **[0.32, 1.47]**	47 39.81 [0.20, 4.34]
$\nu_{pc} = 0.3$	96 20.07 [0.09, 0.92]	55 56.03 [0.19, 0.99]	39 32.48 [0.29, 0.97]	14 5.54 [0.22, 1]

4.2 Finite strain

In this section, we run different benchmark problems to study convergence and stability of different elements using displacement-based formulation and mixed $\boldsymbol{u} - p$ formulations in compressible and nearly incompressible regimes. For the sake of brevity, we consider only Neo-Hookean hyperelastic model with volumetric energy form

$$\psi_{\text{vol}} = kU(J) = \frac{k}{2}\left(J-1\right)^2. \tag{4.4}$$

In order to define the operators for non-symmetric mixed formulation in (3.42), we need first and second derivatives of $U(J)$ i.e.,

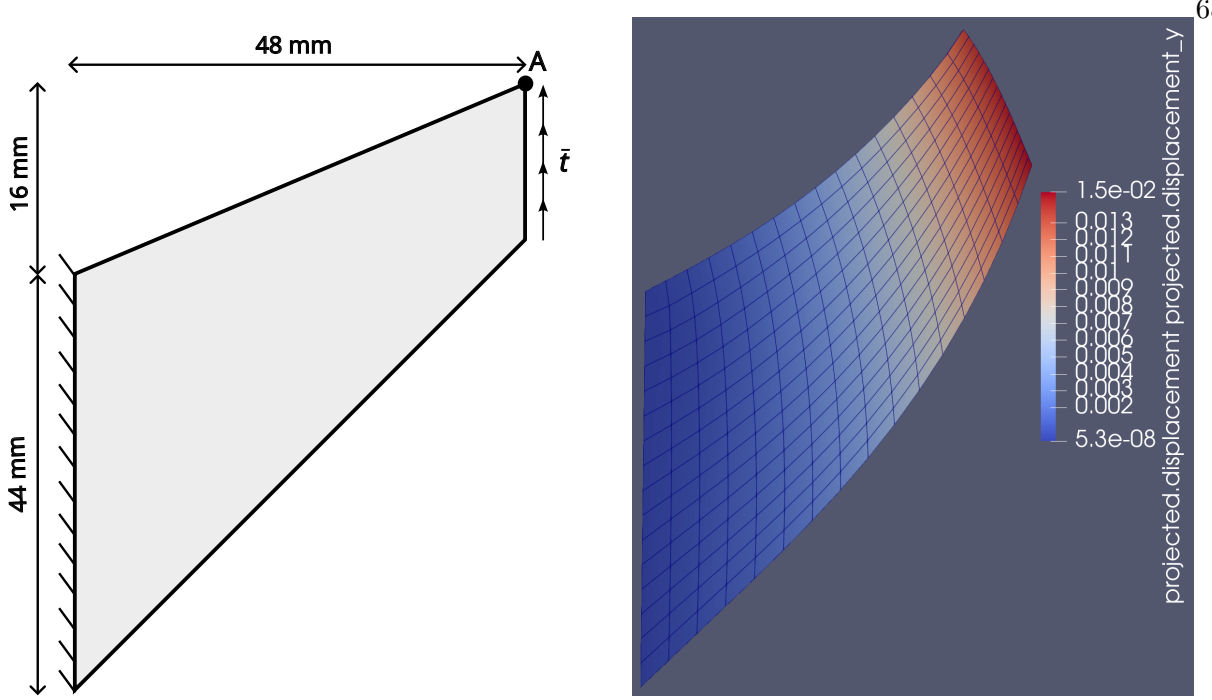
$$U'(J) = J - 1, \qquad U''(J) = 1, \tag{4.5}$$

and for perturbed Lagrange-multiplier method (3.56) (symmetric formulation) we have $\hat{U}(J) = J-1$ and $\hat{U}' = 1$, $\hat{U}'' = 0$.

We solve all problems using quasi-Newton method in 10 pseudo time steps for $t \in [0, 1]$. Each pseudo time step requires a nonlinear solve, which is implemented using PETSc's Scalable Nonlinear Equations Solver (SNES). At each Newton iteration, we use PETSc's Krylov Subspace (KSP), GMRES, to solve (3.85) to a relative tolerance of 10^{-3} with upper triangular block preconditioner (3.94) and multigrid preconditioner (PC) tools.

4.2.1 Plane strain cook's membrane problem

In the first experiment, we follow [67] to test in-plane bending situations in a well-known cook's membrane benchmark. Figure 4.4 displays the geometry and deformed shape of the cook's membrane with linear element (Q_1), Poisson's ratio $\nu = 0.3$ and Young modulus $E = 1.0985$ MPa. To restrict the motion in $x - y$ plane, the slip boundary condition on the grey face is applied in z direction and clamp and traction $\bar{\boldsymbol{t}} = [0, 62.5, 0]^T$ kPa/m^2 boundary conditions are applied on the left and right edges, respectively. We consider number of elements $n_x = n_y = 2^{l+1}, n_z = 1$ in x, y and z directions, with $l \in \{0, 1, ..., 5\}$ indicating the refinement level.

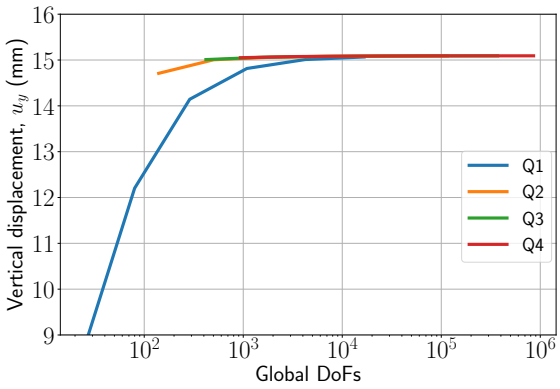


(a) (b)

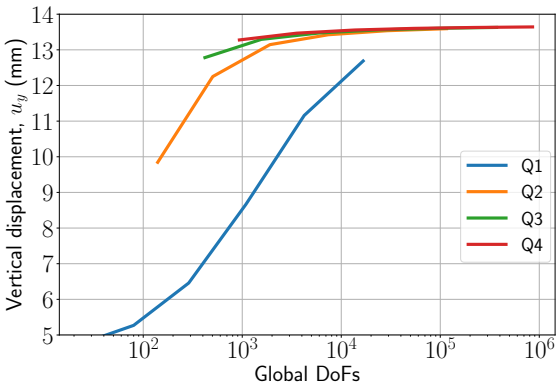
Figure 4.4: Cook's membrane test: (a) geometry and boundary conditions and (b) vertical displacement for $\nu = 0.3$ with single field Neo-Hookean model at refinement level $l = 3$.

Convergence study for vertical displacement of point A is plotted in Figure 4.5 for Poisson's ratios $\nu = 0.3$ and $\nu = 0.498$ and $\nu_p = \nu_{pc} \in \{-1, 0, 0.29\}$ for single field formulation using Q_n, $n = 1, ..., 4$ elements, non-symmetric and symmetric mixed formulations using mixed elements with continuous and discontinuous pressure space i.e., Q_2Q_{1c}, Q_3Q_{2c} and $Q_1P_0, Q_2P_0, Q_2P_1, Q_3Q_1$, respectively (Figure 4.2 displays the 2D case of these elements). The vertical displacement of point A in all the compressible models approach to 15.09 mm, while in incompressible regime u_y is 13.65 mm. It should be noted that results given in [67] is slightly smaller (14.8 mm for compressible and 12 mm for exact incompressible) since they measure the vertical displacement of mid-point at the right edge of the cook's membrane. In single field case, Q_1 element shows the poorest performance and approaches to final vertical displacement with very refined mesh in compressible case, whereas for $\nu = 0.498$ it fails to reach 13.65 mm due to locking effect. As expected, by increasing the

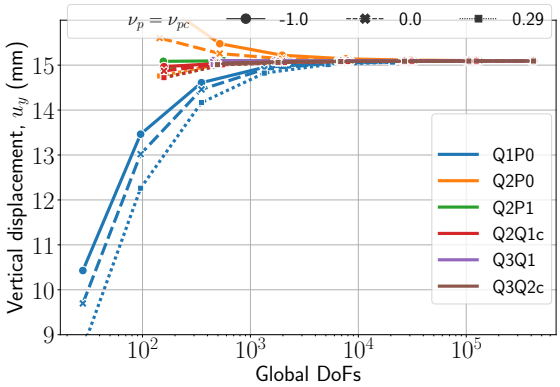
polynomial order, we reach the desire displacement value with a coarser mesh (less DoFs). In both mixed models, Q_1P_0 with all ν_p values and Q_2P_0 with $\nu_p = -1, 0$ start with huge error in displacement when coarse mesh is used (refinement level $l \leq 3$). On the other hand, all other mixed elements and Q_2P_0 with $\nu_p = 0.29$ give accurate results at the early stage of refinement. We run above experiment with Q_5 element, and Q_4Q_{3c} with $\nu_p = 0.29$ using single and mixed fields at refinement level $l = 5$ and considered their results as a reference solution to plot the relative error in displacement as shown in Figure 4.6. It is clear that we lose accuracy in displacement-based formulation and the rate of error degrade as $\nu \longrightarrow 0.5$ especially with linear element Q_1. However, in mixed formulation the rate is almost remains constant. It should be noted that all of the benchmark problems for hyperelasticity in this section have singularity. Thus, we will not see the expected convergence order that we showed in linear case (see Table 4.1, Table 4.2, and Table 4.3).



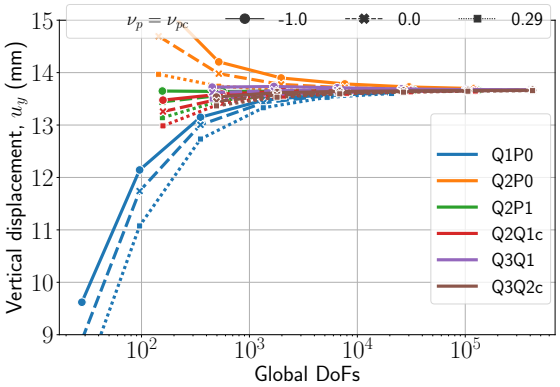
(a) Single field model $\nu = 0.3$.



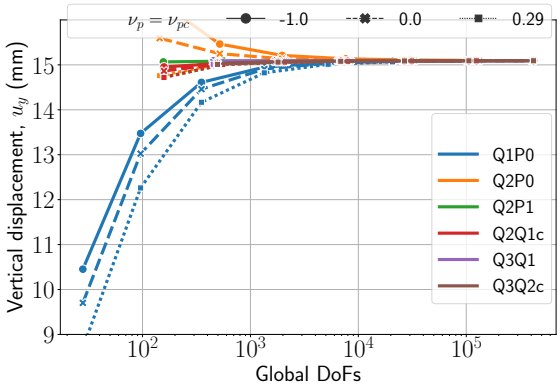
(b) Single field model $\nu = 0.498$.



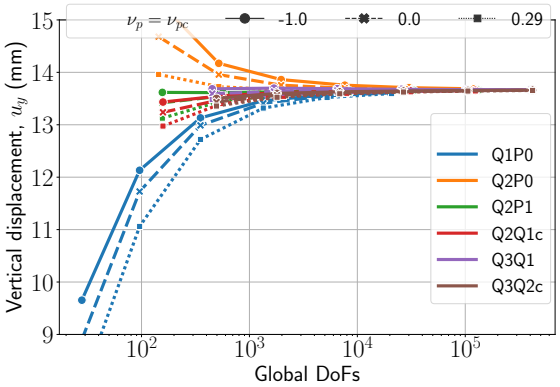
(c) Non-symmetric mixed model $\nu = 0.3$.



(d) Non-symmetric mixed model $\nu = 0.498$.

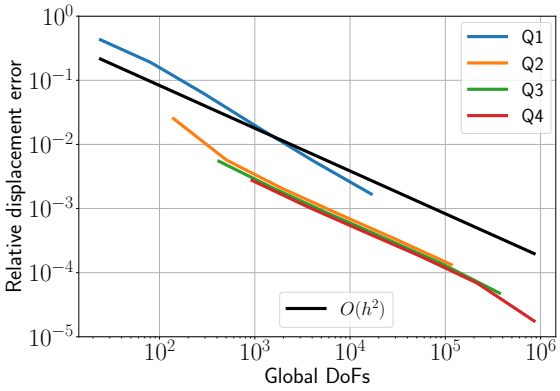


(e) Symmetric mixed model $\nu = 0.3$.

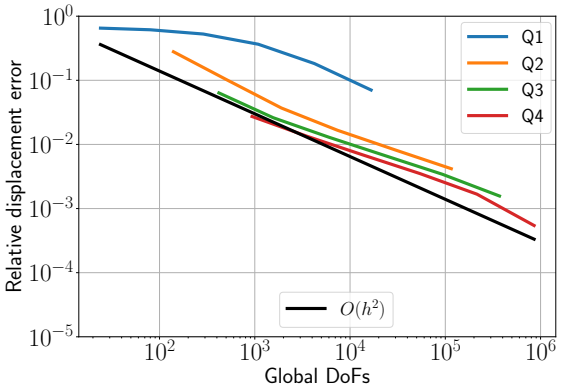


(f) Symmetric mixed model $\nu = 0.498$.

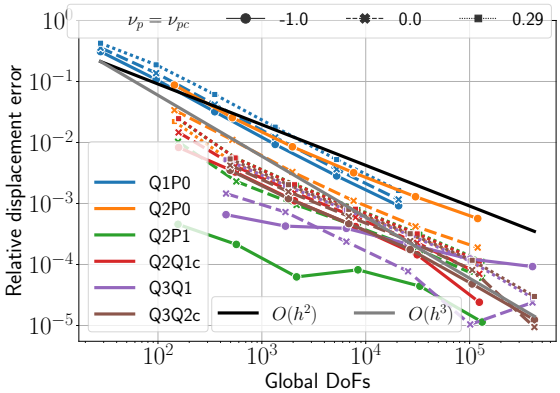
Figure 4.5: Cook's membrane convergence study plot for Neo-Hookean model in compressible and incompressible regime using single field formulation (top row), non-symmetric mixed formulation (middle row), and perturbed Lagrange-multiplier approach (bottom row).



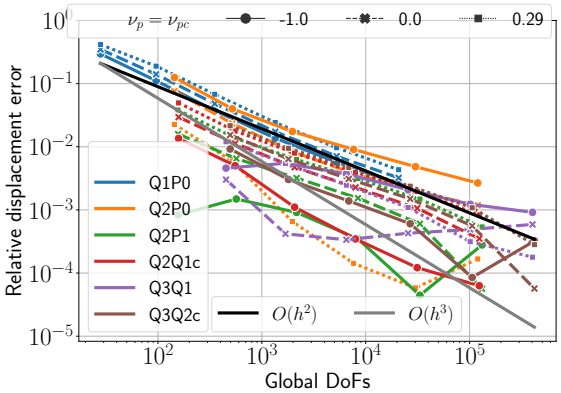
(a) Single field model $\nu = 0.3$.



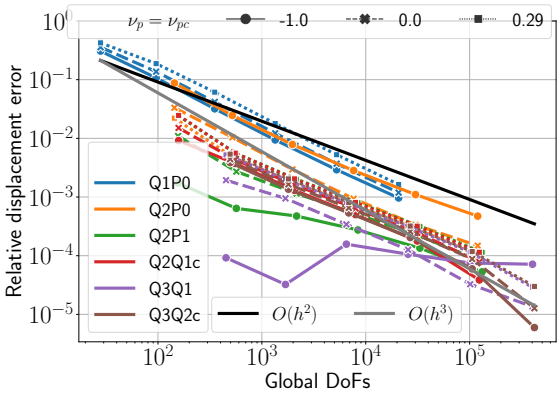
(b) Single field model $\nu = 0.498$.



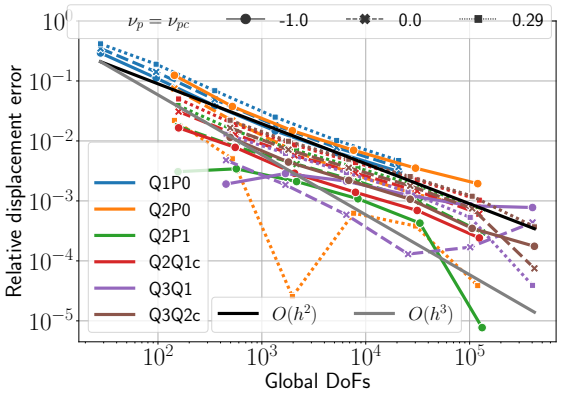
(c) Non-symmetric mixed model $\nu = 0.3$.



(d) Non-symmetric mixed model $\nu = 0.498$.



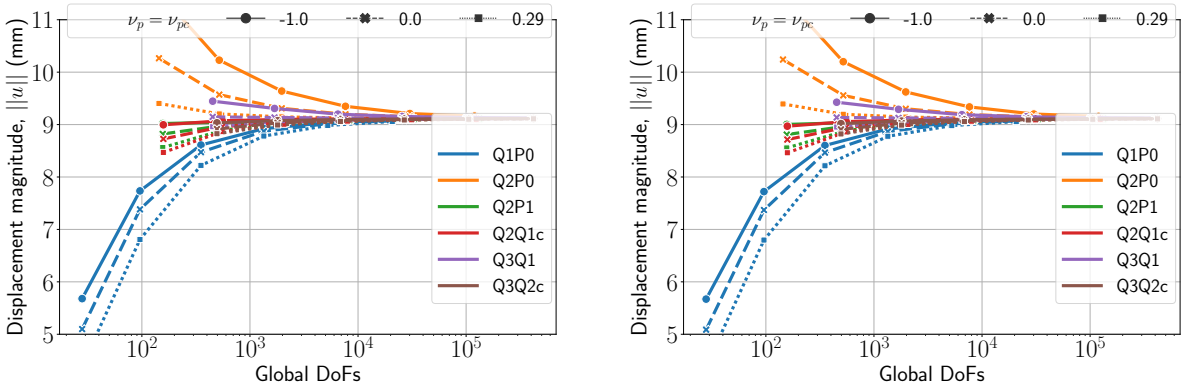
(e) Symmetric mixed model $\nu = 0.3$.



(f) Symmetric mixed model $\nu = 0.498$.

Figure 4.6: Relative displacement error for cook's membrane with Neo-Hookean model in compressible and incompressible regime using single field formulation (top row), non-symmetric mixed formulation (middle row), and perturbed Lagrange-multiplier approach (bottom row).

We repeat our experiment with $E = 240.566$ MPa, $\nu = 0.4999$, and applied traction $\bar{\boldsymbol{t}} = [0, 6250, 0]^T$ kPa/m^2, to verify the displacement magnitude of point A as given in [15]. Figure 4.7 depicts the convergence study for both symmetric and non-symmetric formulations. Similar behavior as discussed above occurred and all mixed elements approach to 9 mm, which is exactly the same value reported by [15].



(a) Non-symmetric mixed model $\nu = 0.4999$.

(b) Symmetric mixed model $\nu = 0.4999$.

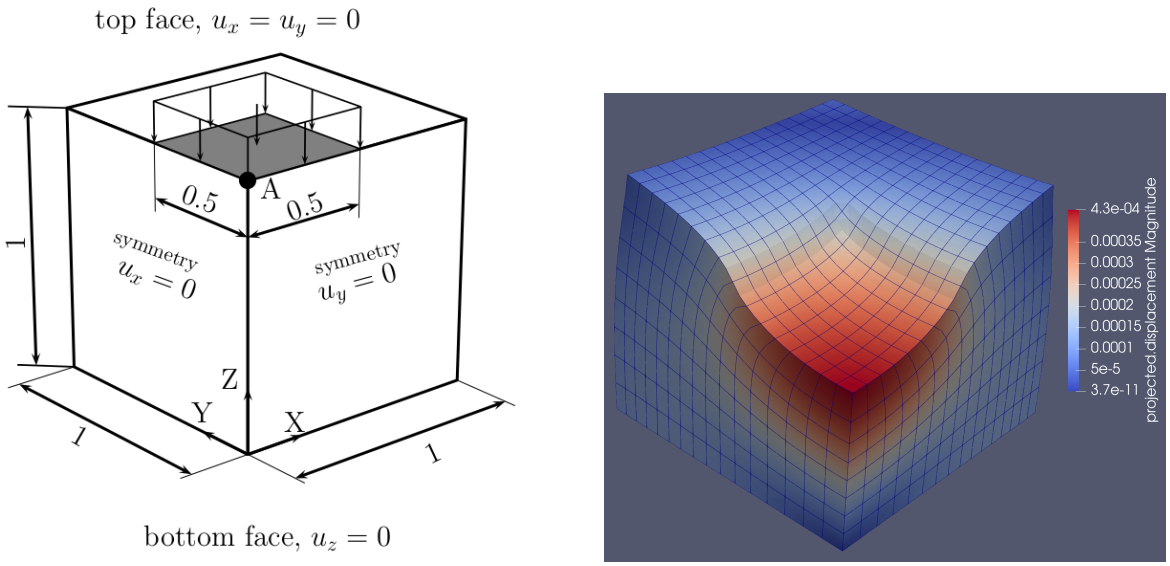
Figure 4.7: Cook's membrane convergence study plot with Neo-Hookean model in incompressible regime using non-symmetric mixed formulation (left), and perturbed Lagrange-multiplier approach (right).

4.2.2 Punch Test

In the previous experiment we only checked the accuracy of the displacement field. However, to check the pressure results, we run a benchmark problem with given boundary conditions described in Figure 4.8 (see [53]), Young's Modulus $E = 240.566$ MPa, Poisson's ratio $\nu = 0.3$ for compressible case and $\nu = 0.499, 0.49999$ for incompressible case using single and mixed formulations. The pressure load applied on the grey area on the top face with magnitude 160 MPa and the discretization is $(n_x, n_y, n_z) = 2^l(4, 4, 2)$ with $l \in \{0, 1, ..., 4\}$ refers to refinement level. The deformed shape on the right side of figure Figure 4.8 at refinement $l = 2$ shows the 43% com-

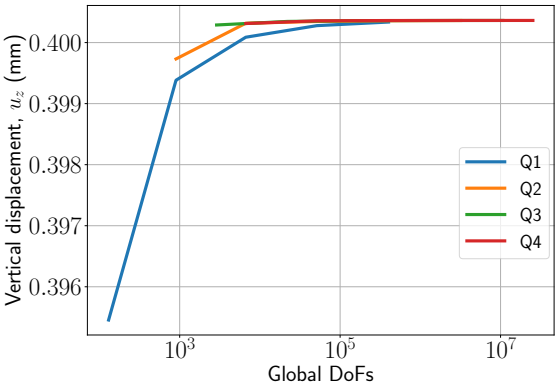
pression with Q_2P_0 element discretization. The convergence study for displacement and pressure with single and mixed models are exhibited in Figure 4.9 and Figure 4.10. In compressible regime vertical displacement of point A approaches 0.4 mm and in incompressible limit the final displacement value is 0.425 mm which is exactly the same as given in [53] for $p_0 = 160$ MPa. Similar to cook's membrane problem, Q_1 element loses accuracy in incompressible case and Q_3, Q_4 elements need at least one level of refinement to reach the final displacement value, while in compressible case for $l = 0$ give the accurate result. The results for both mixed models are similar and with $\nu_p = \nu_{pc} = 0.29$ we get better accuracy at the early stage of the refinement for both compressible and incompressible regimes. The hydrostatic pressure for compressible case reach 76 MPa, which is almost half of the applied pressure p_0. The linear elements Q_1, Q_1P_0 do not converge to the accurate hydrostatic pressure even at the finest level of mesh refinement. Q_2P_0 element gives accurate results with $\nu_p = 0.29$ for compressible case but in the incompressible limit, its result has a huge error for $\nu_p = -1, 0$. For $\nu = 0.499$ with single field formulation, we can see Q_1 element has a huge pressure value due to locking and all other high order elements depicts a higher value for pressure. However, both mixed formulations approach p_0 as expected for $\nu = 0.49999$ without any locking. It should be noted that Q_1P_0 and Q_2P_0 need more refinement to get closer to the expected final pressure value since we interpolate pressure field with constant. On the other hand, by increasing the order of pressure space we reach the 160 MPa after few level of mesh refinement. It seems for element with continuous pressure field Q_3Q_{2c} we get closer to final pressure value with all ν_p values in compressible limit, whereas in incompressible regime $\nu_p = -1$ gives more accurate value. On the contrary, Q_3Q_1 element reach 160 MPa when $\nu_p = 0.29$. Note that Q_3Q_{2c} element uses quadratic element for pressure field thus at the early stage of the mesh refinement it approaches p_0. Figure 4.11 shows the linear iteration count for single field and mixed formulations. Q_1 element uses AMG and its iteration count is increasing under mesh refinement. Q_2, Q_3 and Q_4 shows an excellent performance with maintaining iteration count 12 under mesh refinement in compressible case. However, it increases to 100-150 in the incompressible regime due to bigger condition number of the linear system. In mixed models all the elements have iteration count under 25 and almost

remains constant under mesh refinement in both compressible and incompressible limits, which shows the robustness of our proposed upper triangular preconditioner. However, iteration count of Q_1P_0 element for $\nu = 0.49999$ is increasing due to huge condition number of the matrix since it is not inf-sup stable element, which means an eigenvalue of the system is approaching zero as shown in Figure 4.3.

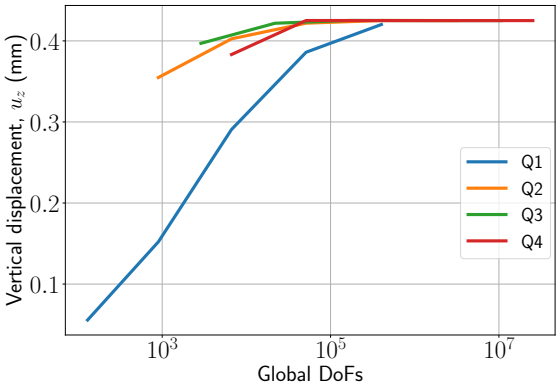


(a) (b)

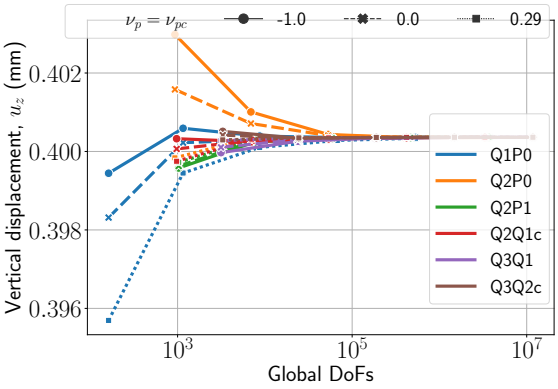
Figure 4.8: 3D block: (a) geometry and boundary conditions described in [53] and (b) deformed shape under compression load $p_0 = 160$ MPa and $\nu = 0.49999$ at refinement level $l = 2$.



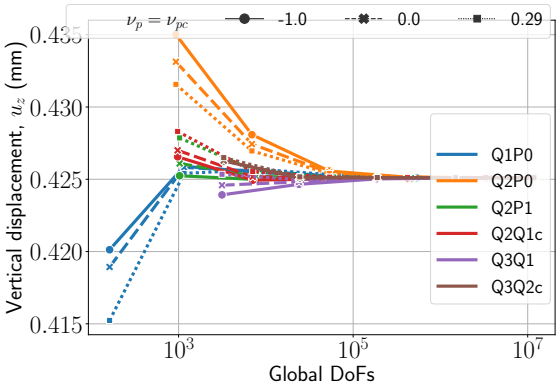
(a) Single field model $\nu = 0.3$.



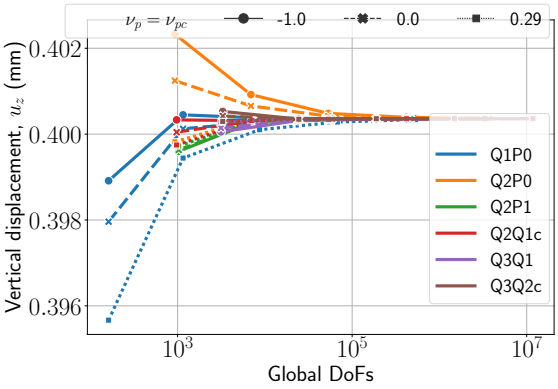
(b) Single field model $\nu = 0.499$.



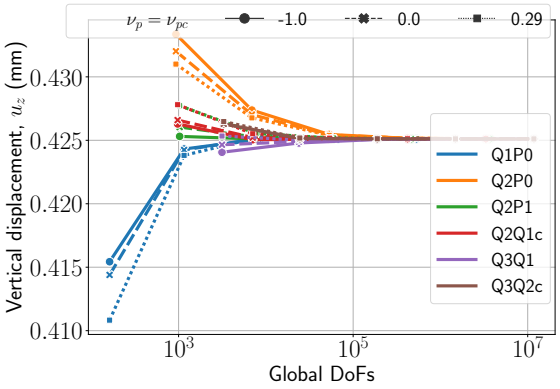
(c) Non-symmetric mixed model $\nu = 0.3$.



(d) Non-symmetric mixed model $\nu = 0.49999$.

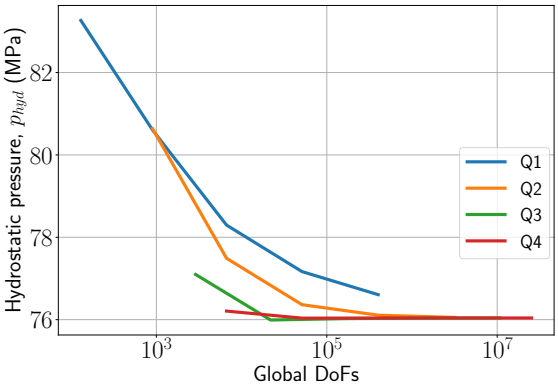


(e) Symmetric mixed model $\nu = 0.3$.

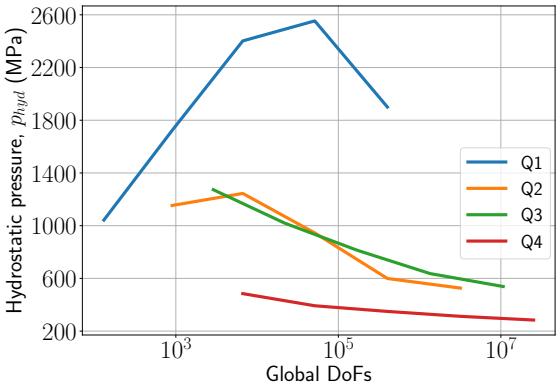


(f) Symmetric mixed model $\nu = 0.49999$.

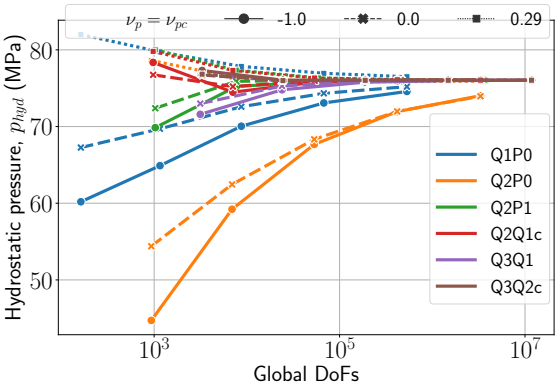
Figure 4.9: Displacement convergence study for the punch test with Neo-Hookean model in incompressible regime using single field formulation (top row), non-symmetric mixed formulation (middle row), and perturbed Lagrange-multiplier approach (bottom row).



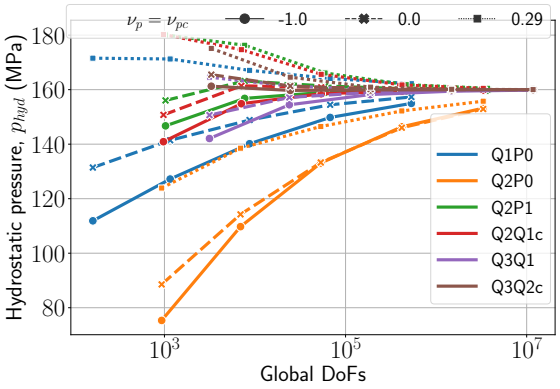
(a) Single field model $\nu = 0.3$.



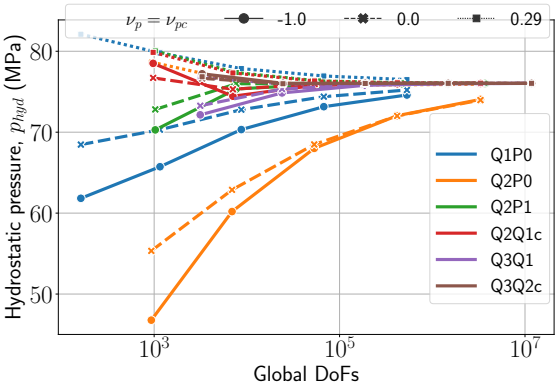
(b) Single field model $\nu = 0.499$.



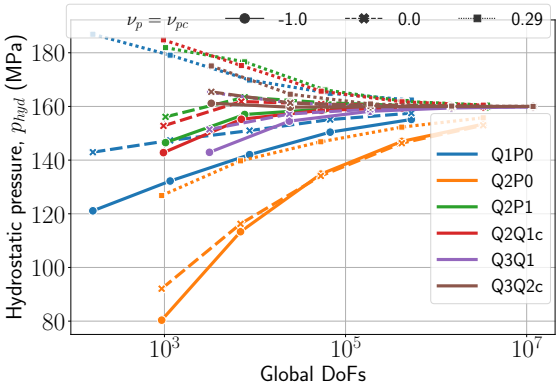
(c) Non-symmetric mixed model $\nu = 0.3$.



(d) Non-symmetric mixed model $\nu = 0.49999$.

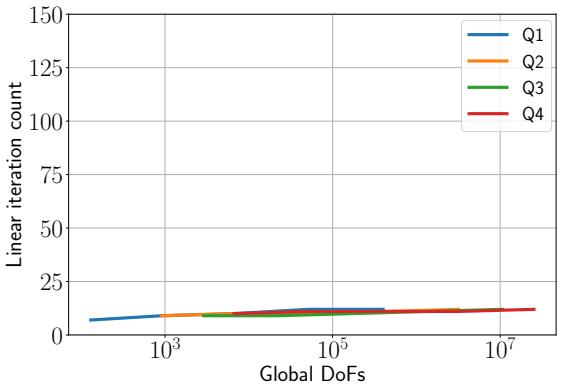


(e) Symmetric mixed model $\nu = 0.3$.

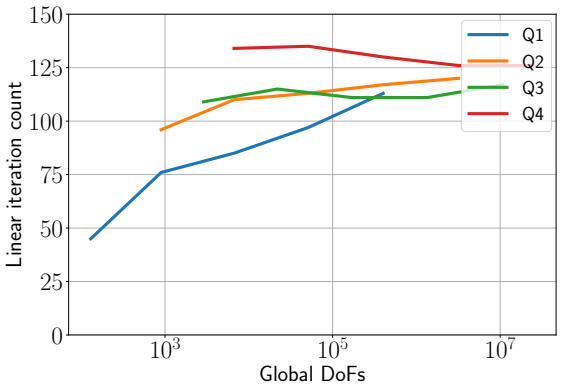


(f) Symmetric mixed model $\nu = 0.49999$.

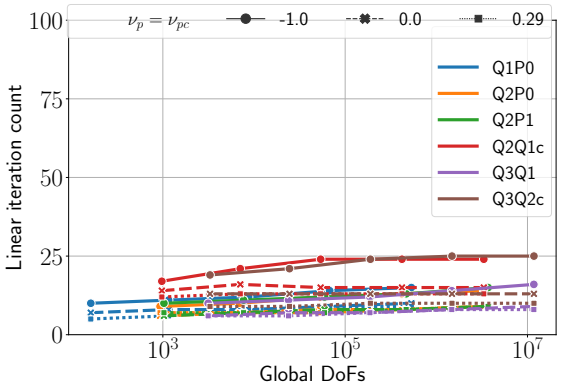
Figure 4.10: Pressure convergence study for the punch test with Neo-Hookean model in incompressible regime using single field formulation (top row), non-symmetric mixed formulation (middle row), and perturbed Lagrange-multiplier approach (bottom row).



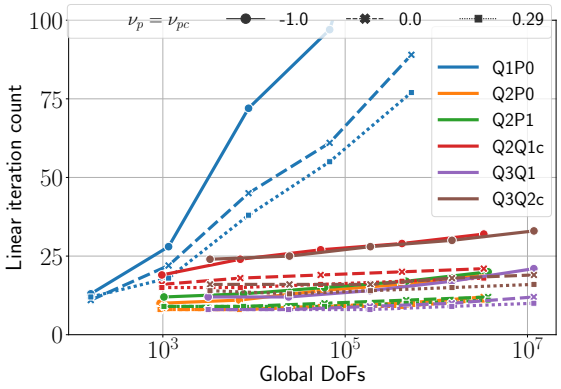
(a) Single field model $\nu = 0.3$.



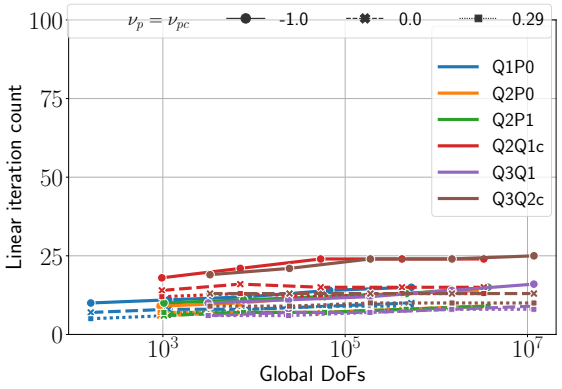
(b) Single field model $\nu = 0.499$.



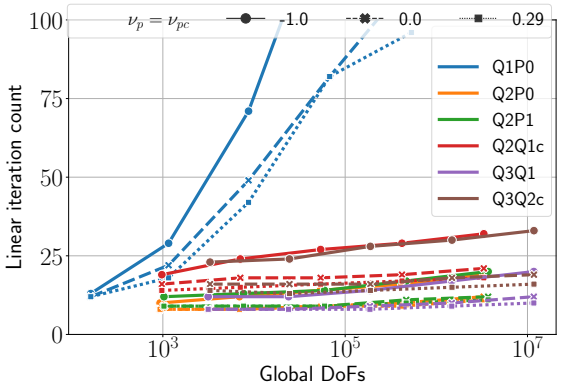
(c) Non-symmetric mixed model $\nu = 0.3$.



(d) Non-symmetric mixed model $\nu = 0.49999$.



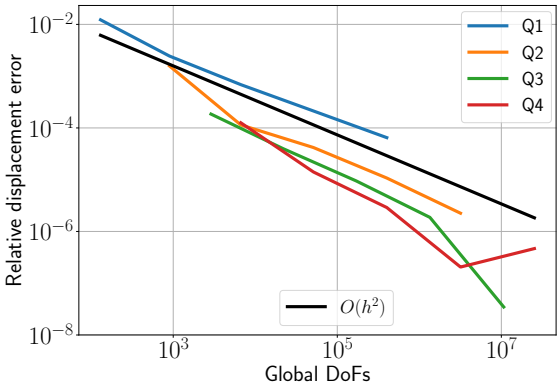
(e) Symmetric mixed model $\nu = 0.3$.



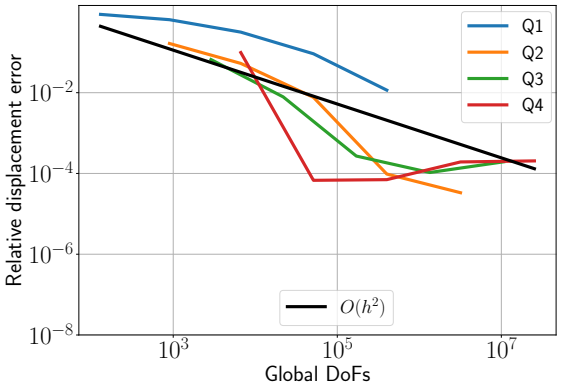
(f) Symmetric mixed model $\nu = 0.49999$.

Figure 4.11: Linear iteration counts for the punch test with Neo-Hookean model in incompressible regime using single field formulation (top row), non-symmetric mixed formulation (middle row), and perturbed Lagrange-multiplier approach (bottom row).

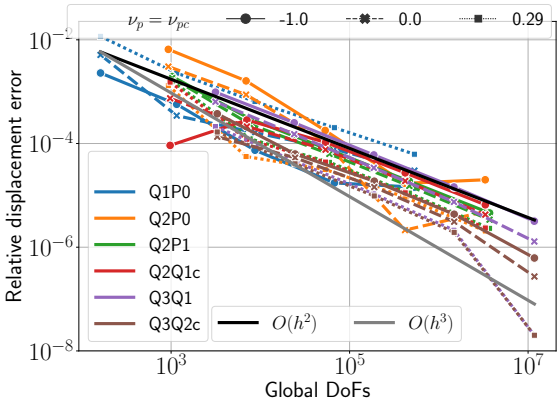
We plot relative displacement and pressure errors in Figure 4.12 and Figure 4.13 using average values of Q_3Q_1 and Q_3Q_{2c} at finest mesh level $l = 4$ and $\nu_p = 0.29$ as a reference solution. For single field, the convergence rate in displacement plot degrade when we have nearly incompressible material since the constant in error estimate (3.5) depends on material properties and pressure relative errors are almost flat and show huge error. In compressible limit, both mixed models give relative error close to 10^{-6} at the finest mesh level similar to single field. However, element like Q_2P_0 with $\nu_p = 0.29$ has close relative error to other high order elements in compressible limit, while for $\nu = 0.49999$, relative error increases. Both higher order mixed element Q_3Q_{2c} and Q_3Q_1 give the smallest relative displacement error with $\nu_p = 0.29$ and $\nu_p = 0$ for compressible and incompressible cases. It should be noted that if we check pressure field for $\nu = 0.49999$, Q_3Q_{2c} with $\nu_p = -1$ has a slightly better accuracy compare to other ν_p values. In general, increasing ν_p results in the lower KSP iteration for all elements (see KSP iteration Figure 4.11). Thus, based on the accuracy and iteration count plots discussed here, we can conclude that $\nu_p \geq 0$ gives more accurate results with smaller KSP iteration and among all the mixed elements, Q_3Q_1 shows the best performance.



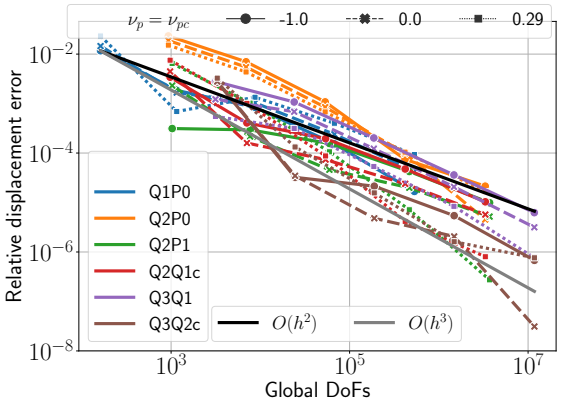
(a) Single field model $\nu = 0.3$.



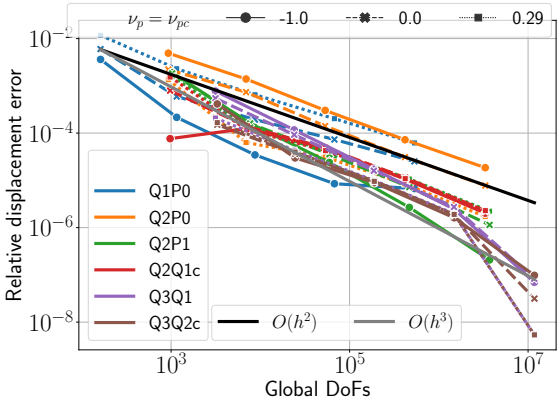
(b) Single field model $\nu = 0.499$.



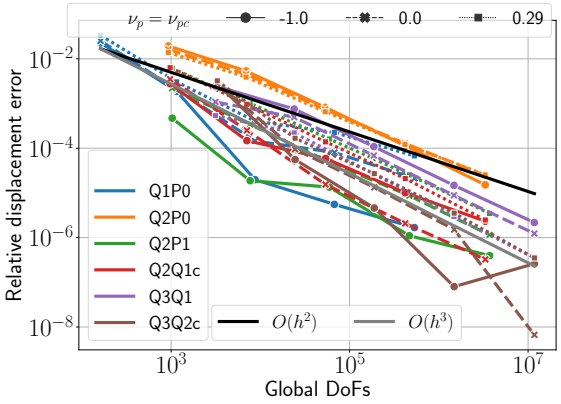
(c) Non-symmetric mixed model $\nu = 0.3$.



(d) Non-symmetric mixed model $\nu = 0.49999$.

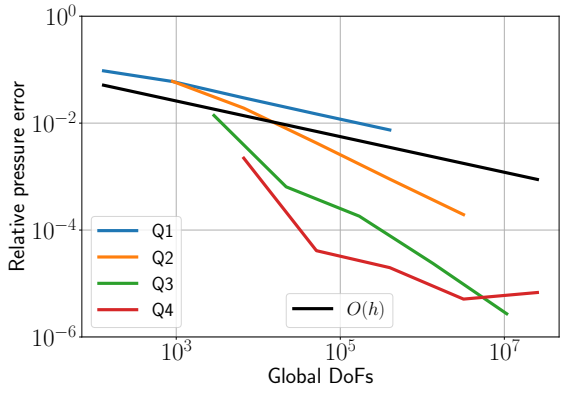


(e) Symmetric mixed model $\nu = 0.3$.

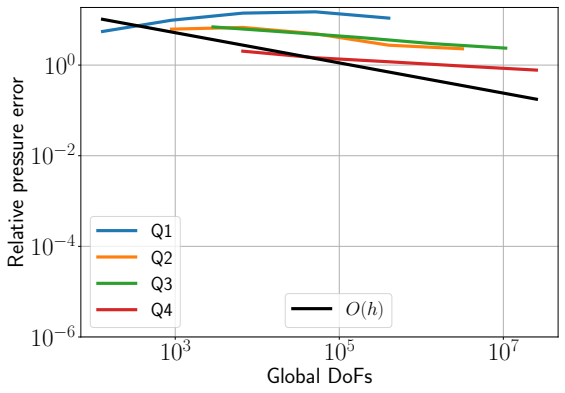


(f) Symmetric mixed model $\nu = 0.49999$.

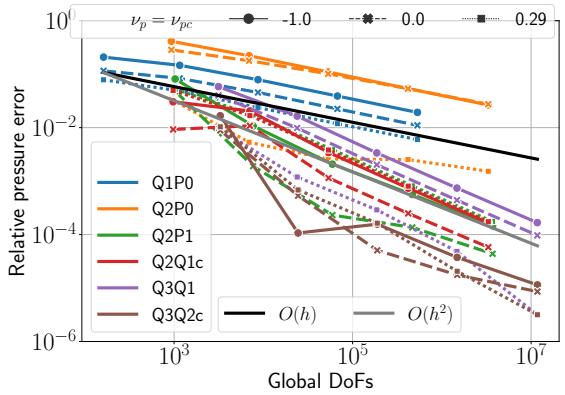
Figure 4.12: Relative displacement error in the punch test with Neo-Hookean model in compressible and incompressible regime using single field formulation (top row), non-symmetric mixed formulation (middle row), and perturbed Lagrange-multiplier approach (bottom row).



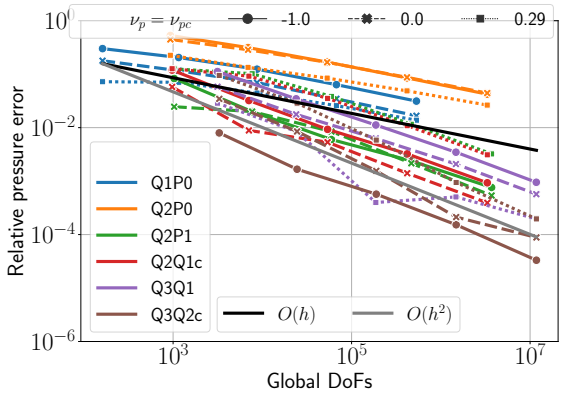
(a) Single field model $\nu = 0.3$.



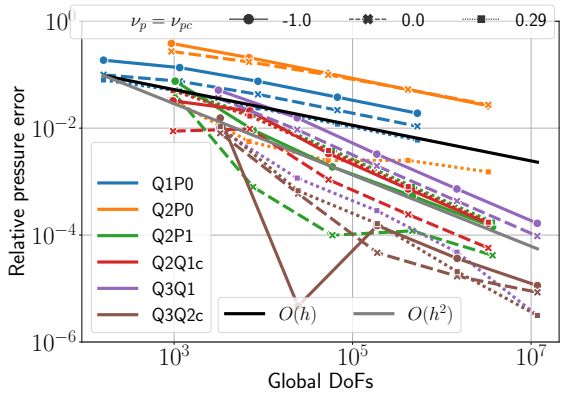
(b) Single field model $\nu = 0.499$.



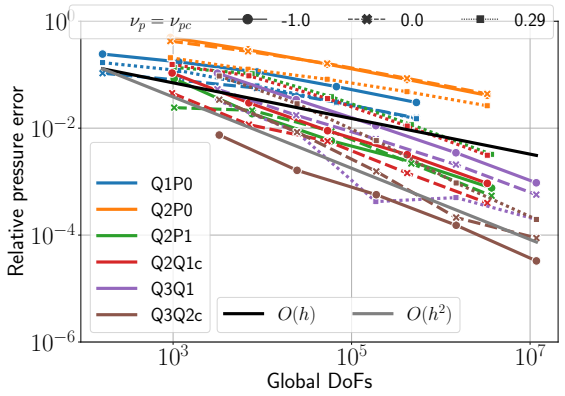
(c) Non-symmetric mixed model $\nu = 0.3$.



(d) Non-symmetric mixed model $\nu = 0.49999$.



(e) Symmetric mixed model $\nu = 0.3$.



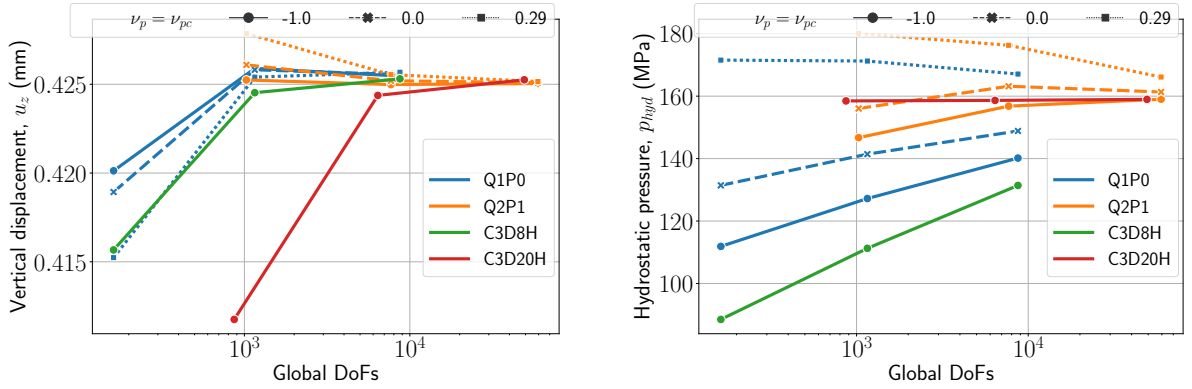
(f) Symmetric mixed model $\nu = 0.49999$.

Figure 4.13: Relative pressure error in the punch test with Neo-Hookean model in compressible and incompressible regime using single field formulation (top row), non-symmetric mixed formulation (middle row), and perturbed Lagrange-multiplier approach (bottom row).

We simulate our experiment in Abaqus by providing user subroutine code (see Appendix E) for Neo-Hookean model with $E = 240.565$ MPa, $\nu = 0.49999$ using linear **C3D8H** and quadratic **C3D20H** elements for mesh refinement levels $l \in 0, 1, 2$. It should be noted that the linear elements in Abaqus and Ratel are the same (8 nodes displacement field and constant pressure), while the quadratic elements have different number of nodes for displacement since Abaqus uses serendipity element (20 nodes in Abaqus versus 27 in Ratel). Table 4.6 shows convergence study obtained by Abaqus and Ratel for the punch test, which are also plotted in Figure 4.14. Displacement results of linear elements C3D8H and Q_1P_0 with $\nu_p = 0.29$ are similar and for quadratic elements, C3D20H has a huge error in displacement at the coarsest mesh (refinement level $l = 0$), while Q_2P_1 is more accurate since it has 27 nodes for displacement. On the other hand, for linear element pressure response of Ratel is closer to applied load $p_0 = 160$ than Abaqus, while Abaqus's pressure response are quite accurate even at the early stage of refinement.

Table 4.6: Abaqus and Ratel results for the punch test with $E = 240.565$ MPa, $\nu = 0.49999$, $p_0 = 160$ MPa, linear and quadratic elements C3D8H and C3D20H, and our Q_1P_0, Q_2P_1 elements for mesh refinement levels $l = 0, 1, 2$.

l	C3D8H		$Q_1P_0(\nu_p = -1)$		$Q_1P_0(\nu_p = 0)$		$Q_1P_0(\nu_p = 0.29)$	
	$u_z(mm)$	$p_{\text{hyd}}(MPa)$	$u_z(mm)$	$p_{\text{hyd}}(MPa)$	$u_z(mm)$	$p_{\text{hyd}}(MPa)$	$u_z(mm)$	$p_{\text{hyd}}(MPa)$
0	0.41567	88.489	0.42013	111.878	0.41893	131.429	0.41524	171.544
1	0.42453	111.258	0.42590	127.206	0.42582	141.424	0.42540	171.248
2	0.42531	131.402	0.42547	140.162	0.42555	148.852	0.42568	167.081
4	–	–	0.42512	154.964	0.42513	157.313	0.42515	162.186
l	C3D20H		$Q_2P_1(\nu_p = -1)$		$Q_2P_1(\nu_p = 0)$		$Q_2P_1(\nu_p = 0.29)$	
	$u_z(mm)$	$p_{\text{hyd}}(MPa)$	$u_z(mm)$	$p_{\text{hyd}}(MPa)$	$u_z(mm)$	$p_{\text{hyd}}(MPa)$	$u_z(mm)$	$p_{\text{hyd}}(MPa)$
0	0.41175	158.475	0.42524	146.715	0.42609	156.045	0.42786	180.070
1	0.42437	158.643	0.42498	156.795	0.42520	163.179	0.42554	176.292
2	0.42525	158.967	0.42504	159.023	0.42509	161.336	0.42515	166.122
4	–	–	0.42511	159.861	0.42511	160.068	0.42511	160.499



(a) Displacement.

(b) Pressure.

Figure 4.14: Displacement (left) and pressure (right) convergence study plots for punch test results obtained by Abaqus and Ratel given in Table 4.6.

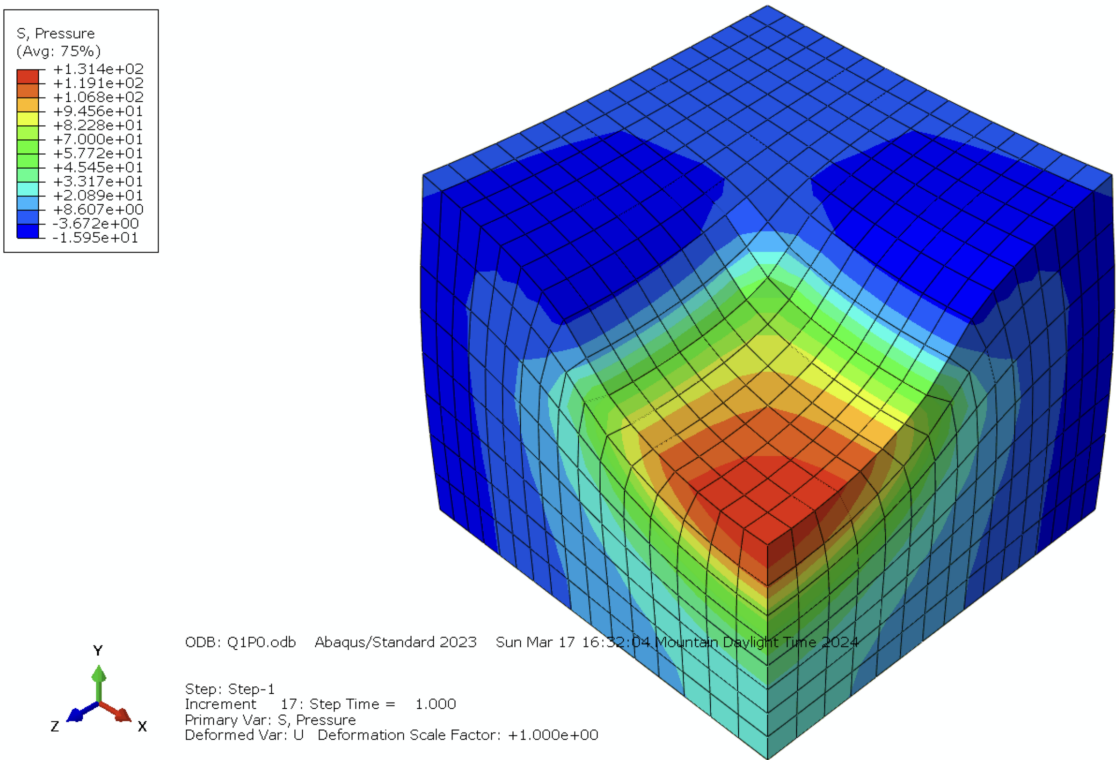
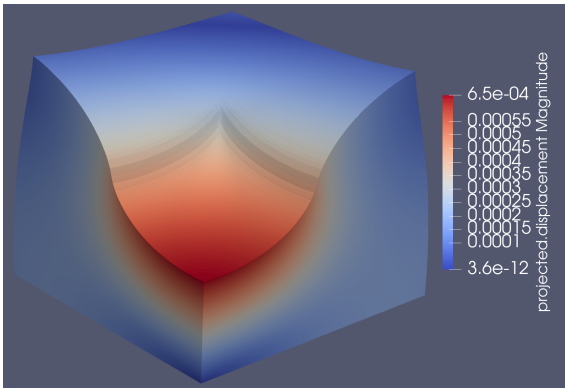


Figure 4.15: Pressure distribution result for the punch test by Abaqus with C3D8H element at refinement level $l = 2$.

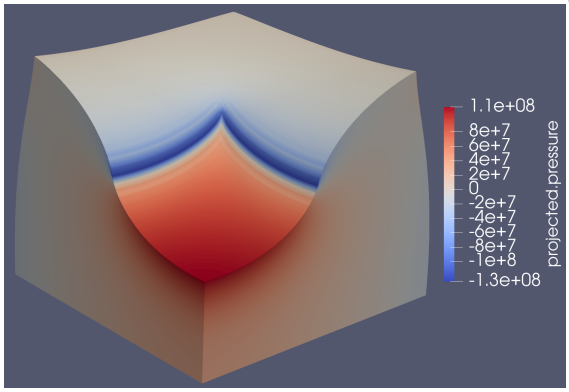
In the next experiment, we run our mesh at refinement level $l = 3$ with applied pressure $p_0 = 320$ MPa, using Q_3 and Q_3Q_{2c} elements with single field and non-symmetric mixed models in compressible $\nu = 0.3$ and incompressible limits $\nu = 0.495$, and $\nu_p = \nu_{pc} = 0.29$. As shown in Figure 4.16, in compressible regime both single and mixed models give similar response with maximum values $u_z = 0.647$ mm, $p_{\text{hyd}} = 128.061$ MPa, and $u_z = 0.647$ mm, $p_{\text{hyd}} = 127.302$ MPa, respectively. In incompressible case as depicted in Figure 4.17, the displacement results for both models are smooth with maximum value $u_z = 0.693$ mm (close to 70% compression as given in [53]). However, using single field model with $\nu = 0.495$ gives a pressure distribution with fluctuation and maximum value $p_{\text{hyd}} = 464.948$ MPa larger than applied load p_0, while the mixed formulation gives smooth pressure response with maximum value $p_{\text{hyd}} = 316.935$ MPa close to applied p_0.

The above experiment is solved in parallel on 8 processors using `avx/blocked` libCEED's

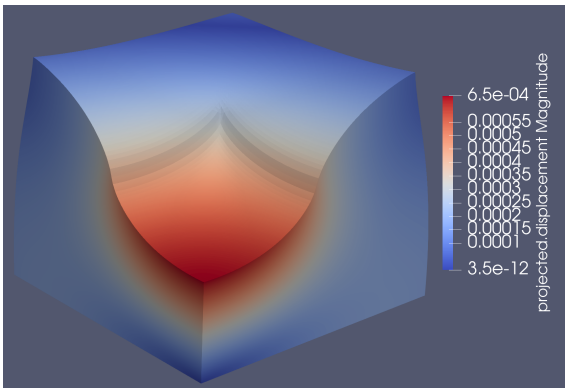
backend on personal laptop in 10 pseudo time steps. Total number of DoFs, SNES iteration count per time step, linear solvers iteration count at each nonlinear step, condition number of the preconditioned operator i.e., $\boldsymbol{P}^{-1}\boldsymbol{J}$ and solve time are given in Table 4.7. When $\nu \longrightarrow 0.5$ the condition number of the system and as a result linear and nonlinear iteration counts and the solve time increase for displacement-based formulation. However, condition number for mixed formulation decreases for $\nu = 0.495$ and iteration count for nonlinear and linear solvers remains almost constant. It should be noted that in incompressible case, the solve time for single field model almost is more than twice of the solve time of mixed formulation even though the size of the system is smaller.



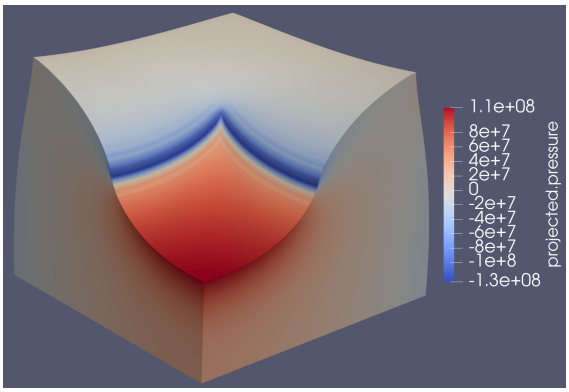
(a) Single field model, displacement distribution



(b) Single field model, pressure distribution

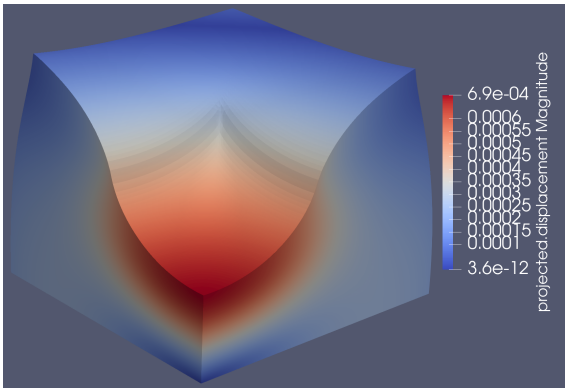


(c) Mixed fields model, displacement distribution

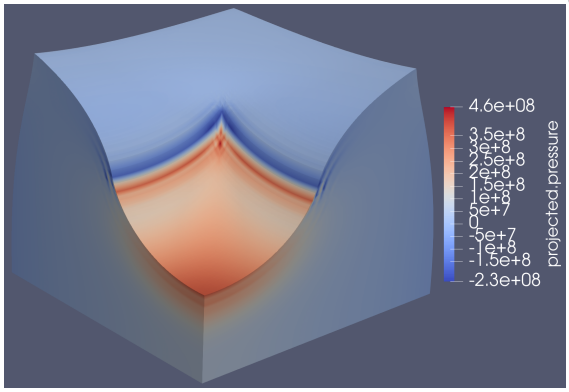


(d) Mixed fields model, pressure distribution

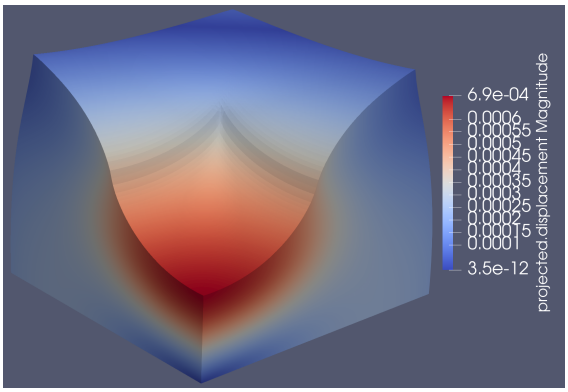
Figure 4.16: Displacement and pressure distribution for the punch test with $p_0 = 320$ MPa and mesh refinement level $l = 3$ obtained by solving displacement-based and mixed non-symmetric Neo-Hookean model with Q_3 and Q_3Q_{2c} elements and Poisson's ratio 0.3.



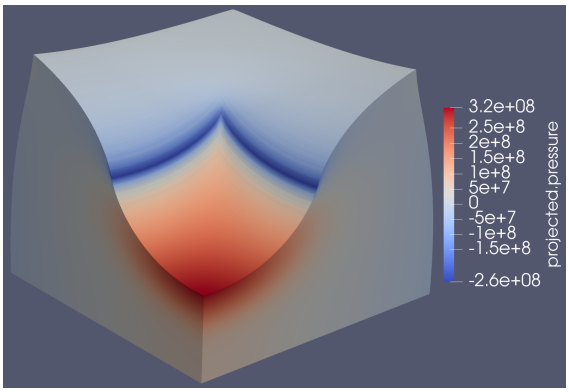
(a) Single field model, displacement distribution



(b) Single field model, pressure distribution



(c) Mixed fields model, displacement distribution



(d) Mixed fields model, pressure distribution

Figure 4.17: Displacement and pressure distribution for the punch test with $p_0 = 320$ MPa and mesh refinement level $l = 3$ obtained by solving displacement-based and mixed non-symmetric Neo-Hookean model with Q_3 and Q_3Q_{2c} elements and Poisson's ratio 0.495.

Table 4.7: Total DoFs, SNES iteration at each pseudo time step, KSP iteration per Newton solver step, condition number of the preconditioned operator, and solve time for the punch test in compressible and incompressible limits displayed in Figure 4.16 and Figure 4.17.

	Q_3, Total DoFs: 1345584			
ν	SNES Its	KSP Its	Condition number	Solve time (sec)
0.3	5	7	3	175.253
0.495	7	30	67	689.826
	Q_3Q_{2c}, Total DoFs: 1485009			
ν	SNES Its	KSP Its	Condition number	Solve time (sec)
0.3	5	8	9	208.126
0.495	5	13	3	289.666

We repeat previous experiment ($E = 240.565$ MPa, $\nu = 0.495$, $\nu_p = 0.29$ and $p_0 = 320$ MPa) with the sparse direct solver using cholesky factorization for approximating $\hat{\mathcal{A}}$ and $\hat{\mathcal{D}}$ in non-symmetric mixed formulation and compare the efficiency (DoFs/sec/CPU) with matrix-free approach under mesh refinement in Figure 4.18. The vertical axis in Figure 4.18 is computed by dividing total DoFs by total solve time and 16 processors. For linear element Q_1P_0 we could run our experiment up to refinement level $l = 3$, but other mixed high-order element results obtained for $l = 2$ due to more memory requirement. However, with matrix-free approach we could refine the mesh up to $l = 3$ for all mixed elements. In both cases Q_2P_1 achieve higher efficiency but as we expected, the matrix-free discretizations efficiency is much higher due to less memory usage per DoF. For example, for a system with size 10^4, with sparse direct solver we can solve 42 DoFs per second on one processor, while with matrix-free the efficiency is 40 times higher.

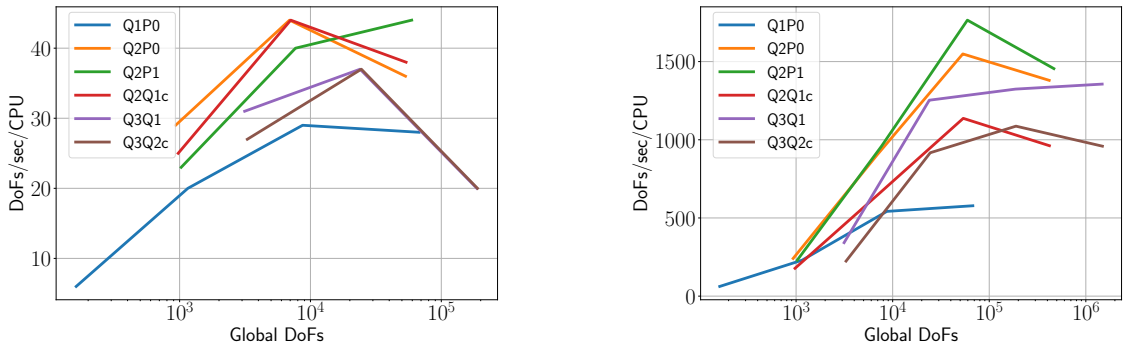


Figure 4.18: Efficiency of the direct solver Cholesky-Cholesky (left) and matrix-free pmg-vpbjacobi (right).

4.2.3 Schwarz Primitive extrusions under compression

Volumetric extrusions of triply periodic minimal surfaces have garnered interest during the additive manufacturing revolution for a range of applications from tissue membranes [54] to metallurgy [3]. We consider the Schwarz Primitive surface, which exhibits interesting geometric and material nonlinearities. Prior finite element analysis of such models [64] using voxelized meshes [63] found that about 30k low order (Abaqus C3D8R) elements were needed to achieve an engineering tolerance of 1%. We consider conformal meshes that attain comparable accuracy with fewer DoF and much fewer elements. To generate such meshes, start with a 24-element 2D manifold mesh of a single unit cell embedded in 3D, replicated to the prescribed extent in each embedding dimension. This mesh is partitioned and distributed using ParMETIS, then refined with new nodes projected to the closest point on the implicit surface

$$\cos 2\pi x + \cos 2\pi y + \cos 2\pi z = 0.$$

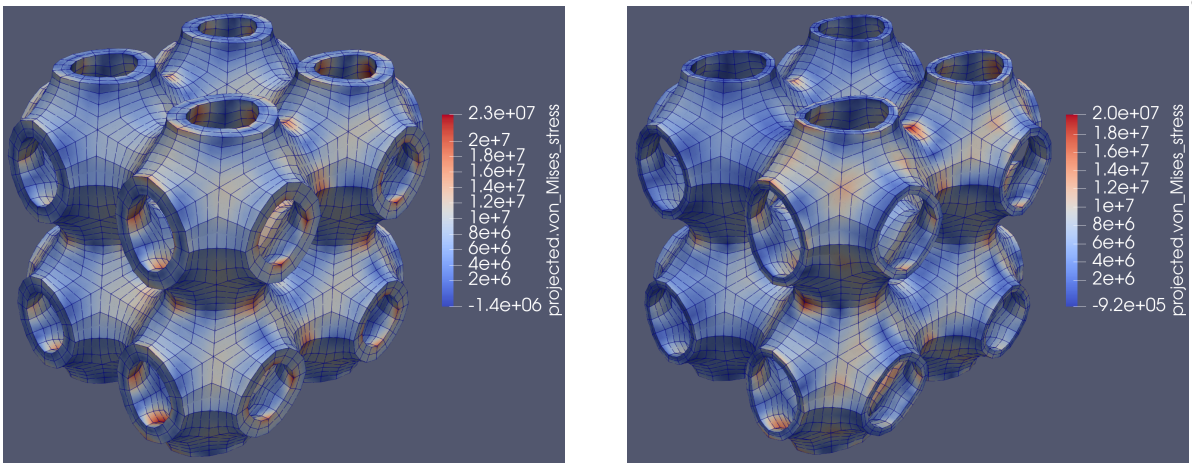


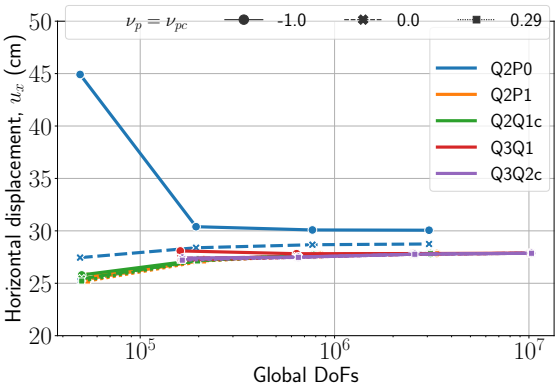
Figure 4.19: Von-Mises distribution for schwarz mesh with refinement level $l = 2$, 2 layers, and extent (2,2,2) using mixed Neo-Hookean model (left) thickness 0.2 under 4 Mpa compression, (right) thickness 0.1 under 2.35 Mpa compression.

The resulting manifold mesh is extruded normal to this surface to the prescribed thickness and number of layers. Figure 4.19 shows a Schwarz Primitive model loaded to about 13% strain on an extent $(2, 2, 2)$ with 2 layers and mesh refinement level $l = 2$. These models, which are available in PETSc-3.17, provide excellent tests for solvers because they exhibit all compressive and bending modes, nonlinearities are activated at local and global scales, coarsening is inherently unstructured, and scaling to a larger domain gives the same accuracy tolerances. We keep the extent, number of layers and strain constant and study the effect of thickness on different mixed elements under mesh refinement $l \in \{1, ..., 4\}$ in compressible and incompressible regimes.

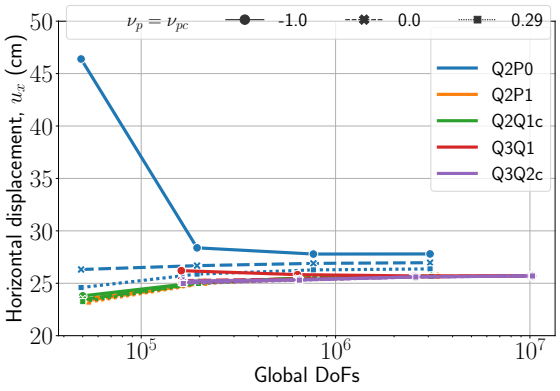
Figure 4.20 shows the displacement for schwarz mesh with thickness 0.2 under 4 MPa compression load for $\nu = 0.3, 0.498$. In compressible limit, Q_2P_0 element with $\nu_p = -1, 0$ yield an slightly different final results and all other elements converge to final horizontal displacement value $u_x = 27.86$ cm. On the other hand, for $\nu = 0.498$, Q_2P_0 loses accuracy even with $\nu_p = 0.29$, which suggest that we should avoid using this element in incompressible regime. The linear iteration count as depicted in Figure 4.21 starts with large number due to higher aspect ration of the element but

under mesh refinement the element becomes almost isotropic, thus all the mixed elements converge to smaller iteration count. When we are in the incompressible regime, the final iteration count is slightly bigger than compressible case due to higher condition number of the system. However, in both cases, iteration count remains under 50, which shows the performance and robustness of our block preconditioner with unstructured mesh discretization. Also, as we expected and shows for linear elasticity in Table 4.5, the discretization with $\nu_p \geq 0$ results in less linear solve iteration for all mixed elements.

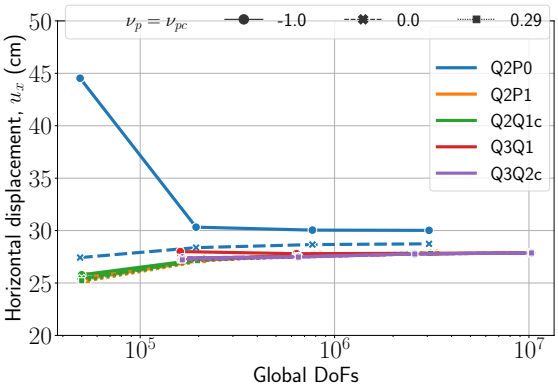
We run our problem using Q_4Q_{3c} element, with $\nu_p = 0.29$ at refined level $l = 4$ as reference solution and plot relative error for displacement field in Figure 4.22. All elements in compressible case have relative error less or equal than 10^{-3} except Q_2P_0 with $\nu_p = -1, 0$. In incompressible regime, Q_2P_0 give inaccurate results for all ν_p cases. For compressible case, Q_3Q_{2c} with $\nu_p = -1$ is more accurate than other elements, however in incompressible limit, Q_3Q_1 with $\nu_p = 0.29$ has the smallest error.



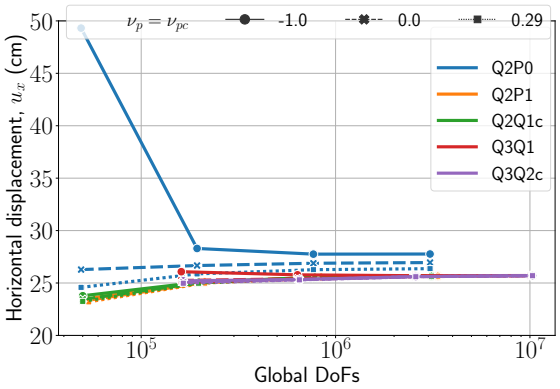
(a) Non-symmetric mixed model $\nu = 0.3$.



(b) Non-symmetric mixed model $\nu = 0.498$.

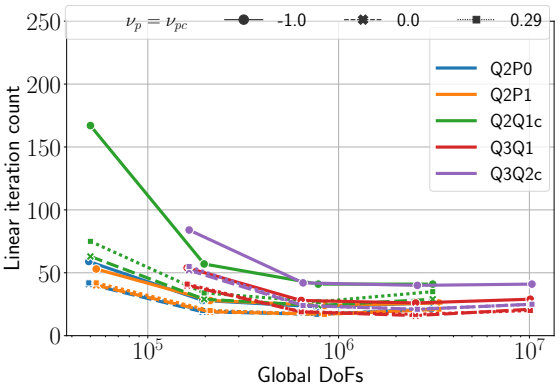


(c) Symmetric mixed model $\nu = 0.3$.

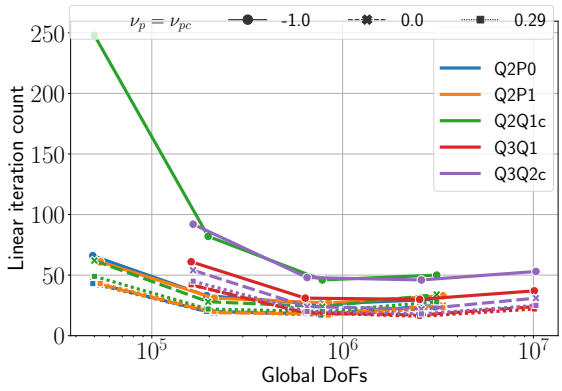


(d) Symmetric mixed model $\nu = 0.498$.

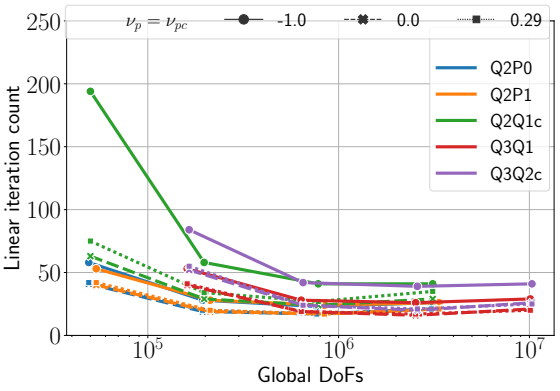
Figure 4.20: Displacement convergence study result for schwarz mesh with thickness 0.2 under 4 MPa compressible load with Neo-Hookean model in compressible and incompressible regimes using non-symmetric mixed formulation (top row), and perturbed Lagrange-multiplier approach (bottom row).



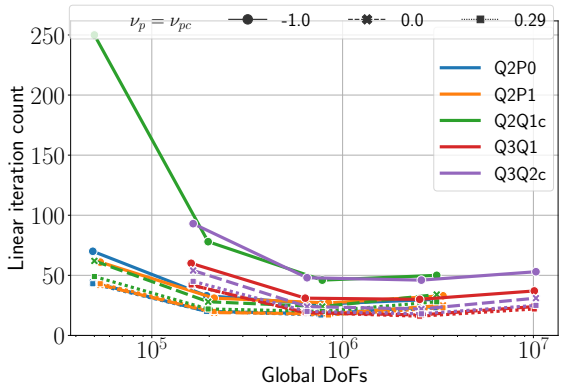
(a) Non-symmetric mixed model $\nu = 0.3$.



(b) Non-symmetric mixed model $\nu = 0.498$.

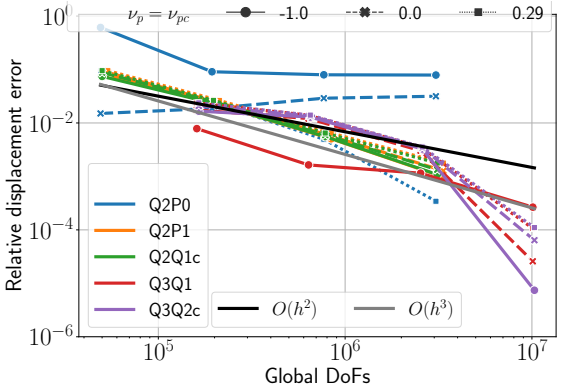


(c) Symmetric mixed model $\nu = 0.3$.

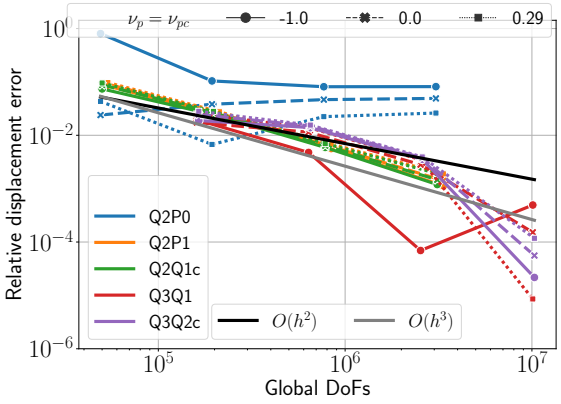


(d) Symmetric mixed model $\nu = 0.498$.

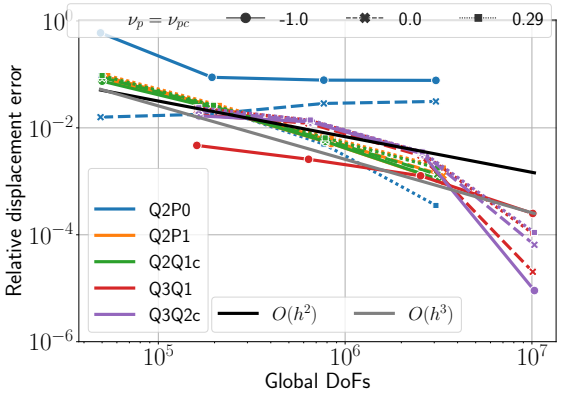
Figure 4.21: Linear iteration count for schwarz mesh with thickness 0.2 under 4 MPa compressible load with Neo-Hookean model in compressible and incompressible regimes using non-symmetric mixed formulation (top row), and perturbed Lagrange-multiplier approach (bottom row).



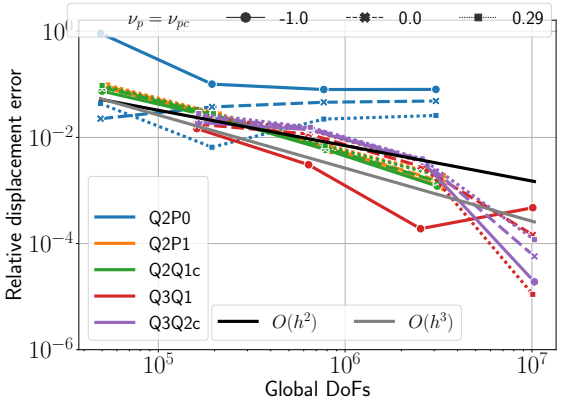
(a) Non-symmetric mixed model $\nu = 0.3$.



(b) Non-symmetric mixed model $\nu = 0.498$.



(c) Symmetric mixed model $\nu = 0.3$.

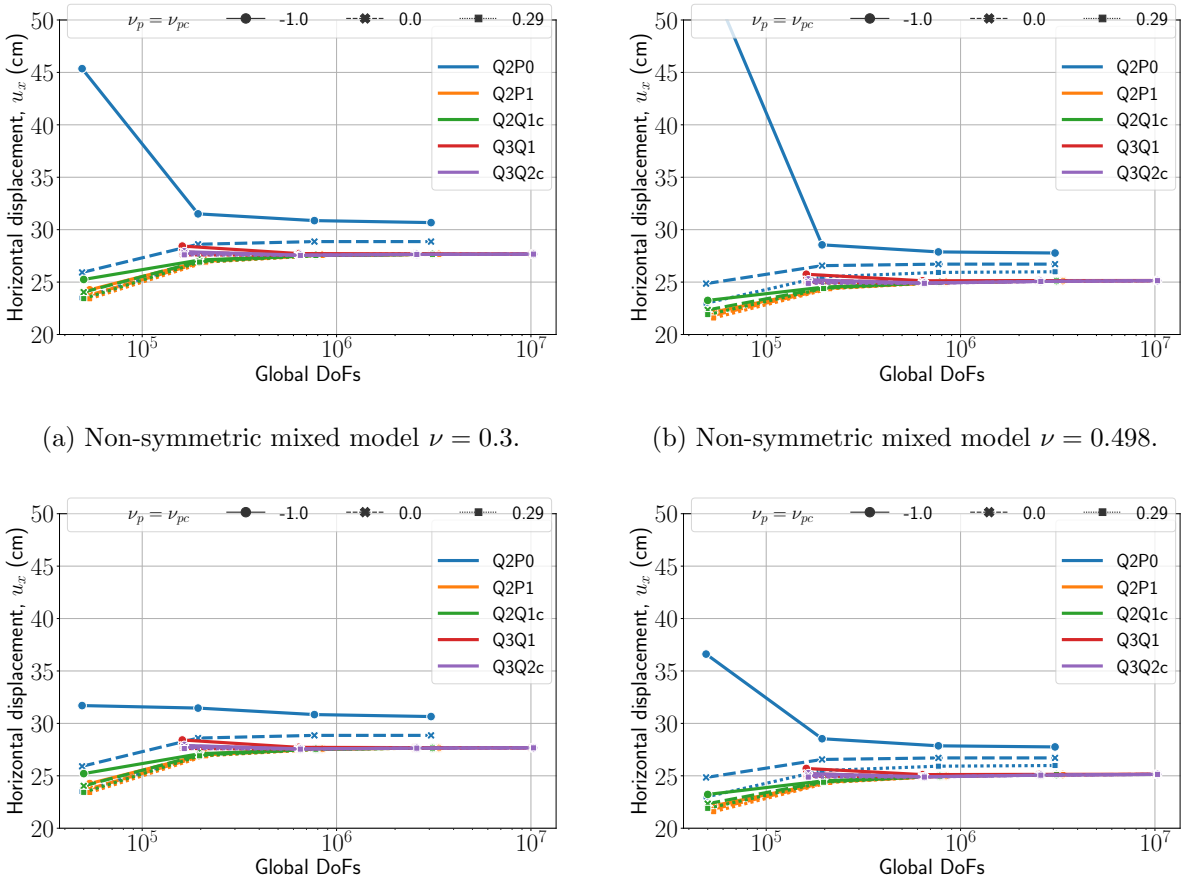


(d) Symmetric mixed model $\nu = 0.498$.

Figure 4.22: Relative error in horizontal displacement for schwarz mesh with thickness 0.2 under 4 MPa compressible load with Neo-Hookean model in compressible and incompressible regimes using non-symmetric mixed formulation (top row), and perturbed Lagrange-multiplier approach (bottom row).

We repeat same experiment with thickness 0.1 as shown in Figure 4.24 and Figure 4.24. Similar to previous case, Q_2P_0 loses accuracy and as expected, the linear iteration count in this case is larger due to higher aspect ratio of the element compare to previous case. Also, under mesh refinement we are getting smaller iteration count since elements become roughly isotropic for $l \geq 3$. Figure 4.25 shows the relative error for thickness 0.1. For compressible case, high order mixed

elements are approaching almost the same relative error for all ν_p except Q_3Q_1 with $\nu_p = -1$. However, in the nearly incompressible case, changing ν_p results in different accuracy and Q_3Q_{2c} with $\nu_p = -1$ and Q_3Q_1 with $\nu_p = 0.29$ are more accurate elements.



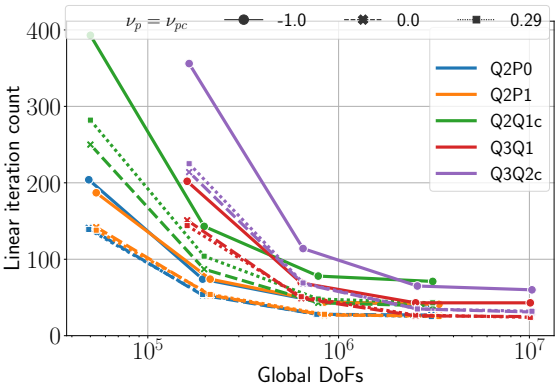
(a) Non-symmetric mixed model $\nu = 0.3$.

(b) Non-symmetric mixed model $\nu = 0.498$.

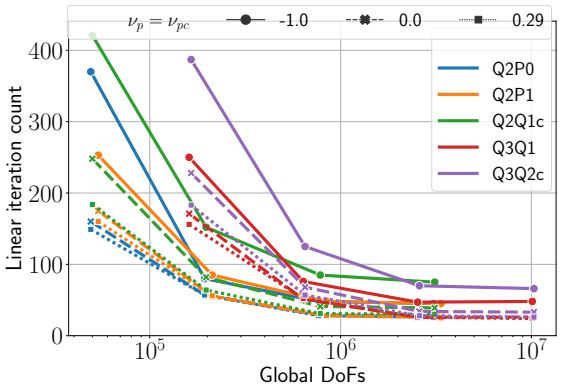
(c) Symmetric mixed model $\nu = 0.3$.

(d) Symmetric mixed model $\nu = 0.498$.

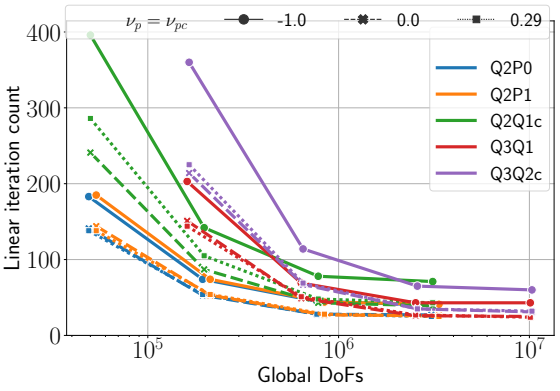
Figure 4.23: Displacement convergence study result for schwarz mesh with thickness 0.1 under 2.35 MPa compressible load with Neo-Hookean model in compressible and incompressible regimes using non-symmetric mixed formulation (top row), and perturbed Lagrange-multiplier approach (bottom row).



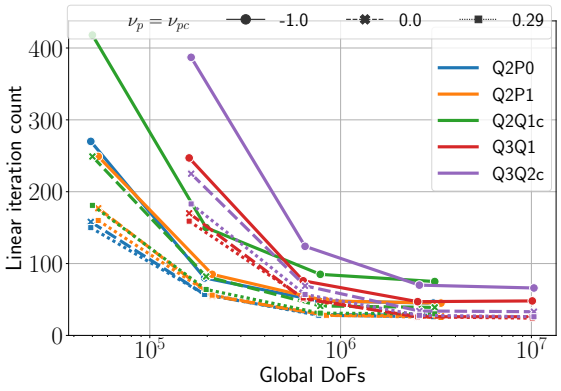
(a) Non-symmetric mixed model $\nu = 0.3$.



(b) Non-symmetric mixed model $\nu = 0.498$.

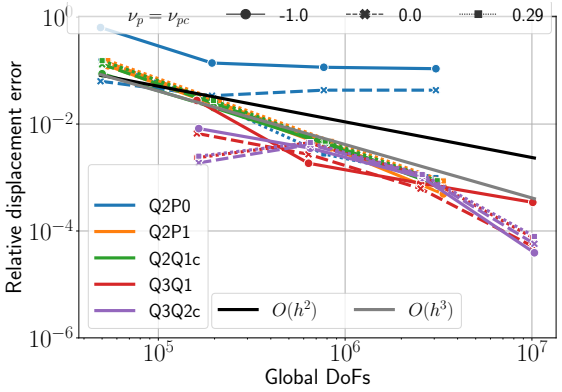


(c) Symmetric mixed model $\nu = 0.3$.

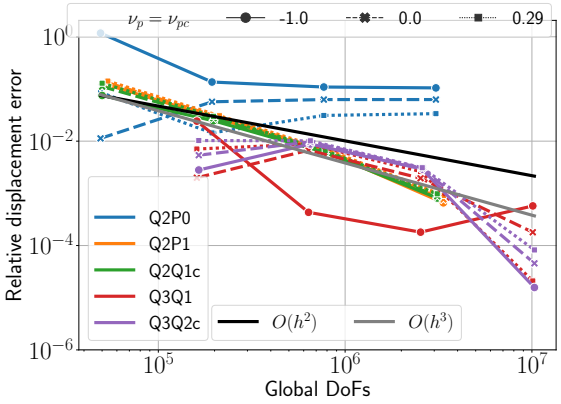


(d) Symmetric mixed model $\nu = 0.498$.

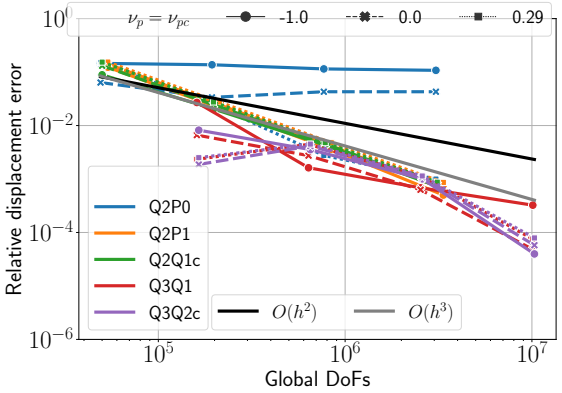
Figure 4.24: Linear iteration count for schwarz mesh with thickness 0.1 under 2.35 MPa compressible load with Neo-Hookean model in compressible and incompressible regimes using non-symmetric mixed formulation (top row), and perturbed Lagrange-multiplier approach (bottom row).



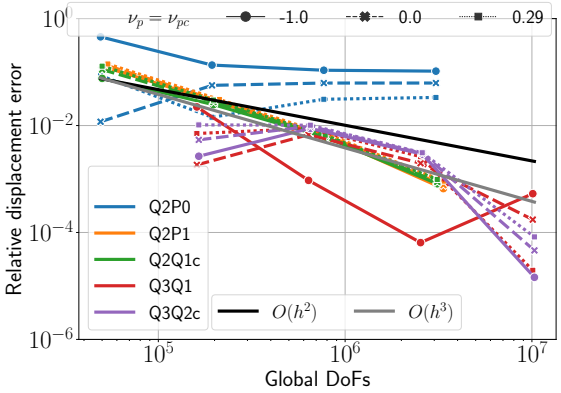
(a) Non-symmetric mixed model $\nu = 0.3$.



(b) Non-symmetric mixed model $\nu = 0.498$.



(c) Symmetric mixed model $\nu = 0.3$.



(d) Symmetric mixed model $\nu = 0.498$.

Figure 4.25: Relative error in horizontal displacement for schwarz mesh with thickness 0.1 under 2.35 MPa compressible load with Neo-Hookean model in compressible and incompressible regimes using non-symmetric mixed formulation (top row), and perturbed Lagrange-multiplier approach (bottom row).

It should be noted that all above experiments in finite strain performed in current configuration formulation section 3.4 due to smaller solve time. Both algorithms for the tangent (Jacobian) operator in initial (see (3.35), (3.34)) and current configurations using non-symmetric formulations are given in algorithms 1 and 2, respectively. Algorithm 1 exploits two static storage containers, geometric factor $\nabla_X \boldsymbol{\xi}$ (9 scalars), and quadrature weight $W = \hat{W} \left| \nabla_\xi \boldsymbol{X} \right|$ (1 scalar), and one com-

puted stored data $\nabla_X \boldsymbol{u}, p$ (10 scalars), which is stored during the residual evaluation step for a total of 20 double precision scalars stored per quadrature point. Algorithm 2 shows current configuration Jacobian evaluation with one static storage W (1 scalar), and computed stored values $\nabla_x \boldsymbol{\xi} = \nabla_X \boldsymbol{\xi}\, \boldsymbol{F}^{-1}$, $\boldsymbol{e}$, $J-1$, $J^{-2/3}$, p (18 scalars) in total 19 double precision scalars stored per quadrature point. Table 4.8 compares total solve time between initial and current configuration on schwarz mesh with extent $(2,2,2)$, thickness 0.2 and refinement level $l = 2$ (193824 DoF), using symmetric and non-symmetric mixed formulations with Q_2P_0 element and $\nu_p = 0.29$ in compressible limit on 8 processors. It is clear that current configuration is about 40 second faster than initial configuration and it is a better choice for large scale problem.

Algorithm 1 Tangent operator in initial configuration stores 10 double precision scalars per quadrature point ($\nabla_X \boldsymbol{\xi}$, W), and computes and stores 9 double precision scalars per quadrature point during residual evaluation ($\nabla_X \boldsymbol{u}$, p)

Input: $\nabla_\xi \,\mathrm{d}\boldsymbol{u}$, $\mathrm{d}p$ plus cached values

for each element $e \in \Omega$ **do**

for each quadrature point q on e **do**

evaluate $\boldsymbol{F} = \boldsymbol{I} + \nabla_X \boldsymbol{u}, \quad J = |\boldsymbol{F}|, \quad J^{-2/3}, \quad \boldsymbol{C}^{-1}, \quad \boldsymbol{E}, \quad \boldsymbol{E}_{\mathrm{dev}}$

evaluate $\boldsymbol{S} = 2\mu J^{-2/3} \boldsymbol{C}^{-1} \boldsymbol{E}_{\mathrm{dev}} + (k_p J U' - pJ)\, \boldsymbol{C}^{-1}$

evaluate $\mathrm{d}\boldsymbol{F} = (\nabla_\xi \,\mathrm{d}\boldsymbol{u})(\nabla_X \xi), \quad \mathrm{d}\boldsymbol{E} = \mathrm{sym}\left(\boldsymbol{F}^T \,\mathrm{d}\boldsymbol{F}\right), \quad \boldsymbol{C}^{-1} : \mathrm{d}\boldsymbol{E}, \quad \mathrm{trace}\,\mathrm{d}\boldsymbol{E}$

evaluate $\mathrm{d}\boldsymbol{C}^{-1} = -2\boldsymbol{C}^{-1}\,\mathrm{d}\boldsymbol{E}\boldsymbol{C}^{-1}$

evaluate $\mathrm{d}\boldsymbol{S} = \mathrm{d}\boldsymbol{S}_{\mathrm{iso}} + \mathrm{d}\boldsymbol{S}^u_{\mathrm{vol}} + \mathrm{d}\boldsymbol{S}^p_{\mathrm{vol}}$

evaluate $\mathrm{d}\boldsymbol{P} = \mathrm{d}\boldsymbol{F}\boldsymbol{S} + \boldsymbol{F}\,\mathrm{d}\boldsymbol{S}$

end for

end for

Algorithm 2 Tangent operator in current configuration stores 1 double precision scalars per quadrature point (W), and computes and stores 9 double precision scalars per quadrature point during residual evaluation ($\nabla_x \boldsymbol{\xi}$, $\boldsymbol{e}$, $J-1$, $J^{-2/3}$, p)

Input: $\nabla_\xi \, \mathrm{d}\boldsymbol{u}$, $\mathrm{d}p$ plus cached values

for each element $e \in \Omega$ **do**

for each quadrature point q on e **do**

evaluate $\boldsymbol{e}_{\mathrm{dev}}$

evaluate $\boldsymbol{\tau} = 2\mu J^{-2/3}\boldsymbol{e}_{\mathrm{dev}} + (k_p J U' - pJ)\,\boldsymbol{I}$

evaluate $\nabla_x \, \mathrm{d}\boldsymbol{u} = (\nabla_\xi \, \mathrm{d}\boldsymbol{u})\,(\nabla_x \xi)$, $\quad \mathrm{d}\boldsymbol{\epsilon} = \mathrm{sym}\,(\nabla_x \, \mathrm{d}\boldsymbol{u})\,,$ $\quad \mathrm{trace}\,\mathrm{d}\boldsymbol{\epsilon}$, $\quad \mathrm{trace}\,\mathrm{d}\boldsymbol{e}$

evaluate $\nabla_x \, \mathrm{d}\boldsymbol{u}\boldsymbol{\tau} + \boldsymbol{F}\,\mathrm{d}\boldsymbol{S}\boldsymbol{F}^T$

end for

end for

Table 4.8: Performance for different Jacobian representations in mixed Neo-Hookean hyperelasticity (around 200 kDoF). Stored values before the semicolon are constant data while those after are a byproduct of residual evaluation.

Problem	Storage	Scalars	Time (s)
current	$W; \nabla_x \boldsymbol{\xi}, \boldsymbol{e}, J-1, J^{-2/3}, p$	19	147.317
initial	$W, \nabla_X \boldsymbol{\xi}; \nabla_X \boldsymbol{u}, p$	20	189.407
current PL	$W; \nabla_x \boldsymbol{\xi}, \boldsymbol{e}, J-1, J^{-2/3}, p$	19	149.188
initial PL	$W, \nabla_X \boldsymbol{\xi}; \nabla_X \boldsymbol{u}, p$	20	182.734

4.3 Conclusion

In this chapter we performed convergence study for single and mixed elasticity in small strain. For $\boldsymbol{u}$-based formulation, element Q_n, where n is polynomial order converge with rate of $n+1$. However, this rate will degrade when $\nu \to 0.5$ since the constant in error estimate depends

on material properties. In mixed linear case, first we analyzed inf-sup constant for different mixed elements with continuous and discontinuous pressure field, then, we compute convergence order for a 2D problem with full Dirichlet boundary condition. For mixed elements with discontinuous pressure, we can have non-tensor and tensor basis for pressure field. The former, are $Q_1P_0, Q_2P_0,$ andQ_2P_1 with convergence order $2, 2, 3$ and $1, 1, 2$ for displacement and pressure field and the latter is Q_3Q_1 with convergence order $3, 2$ for $\boldsymbol{u}, p$. Mixed element with continuous pressure like Q_2Q_{1c}, Q_3Q_{2c} has a tensor basis for pressure field and convergence orders for $\boldsymbol{u}$ and p are $3, 4$ and $2, 3$, respectively. We examined our proposed general mixed formulation with upper-triangular block preconditioner in small strain with full Dirichlet and clamp-traction boundary conditions. In the case of full Dirichlet, the linear iteration count, condition number and range of eigenvalue are almost similar for different ν_p and ν_{pc}. However, when we changed the boundary condition, we observed huge eigenvalue due to presence of dilation null space in displacement block. By Increasing ν_p and ν_{pc} from $-1 \rightarrow 0.3$, results in smaller KSP iteration count, condition number and maximum eigenvalue close to 1.

In finite strain, our formulation for singe and mixed models are written in general form and user can choose any convex function for volumetric energy term to experiment with. We used Neo-Hookean model with quadratic function i.e., $\psi_{\text{vol}} = k(J-1)^2/2$ and compared $\boldsymbol{u}$-based formulation and symmetric and non-symmetric $\boldsymbol{u} - p$ formulations in compressible and nearly incompressible limits with $\nu_p = \nu_{pc} \in -1, 0, 0.29$ and different elements. In the first experiment, we performed convergence study for plain strain cook's membrane problem in the bending mode. We verified our displacement results with previous studies. We depicted that the linear iteration count remains constant under mesh refinement in compressible and incompressible regimes for mixed formulation, which shows robustness and performance of our proposed preconditioner. However, with single field model, when $\nu \rightarrow 0.5$ the KSP iteration is increasing due to larger condition number of the system. In the next experiment, we run the 3D block under compression load in compressible and nearly incompressible limits. Displacement result with single and mixed fields

are similar and converging to same value in compressible case, except Q_1 element which has a huge error for $\nu = 0.499$. Pressure response with $\boldsymbol{u}$-based approaching to same value (except Q_1) as mixed formulation in compressible limit but in nearly incompressible case the final value are larger than applied pressure, which shows the necessity of mixed formulation. All the mixed elements approached to the applied load when $\nu = 0.49999$. However, lower order element like Q_1P_0 or Q_2P_0 need more mesh refinement to decreases the relative error. Moreover, we compared our results with commercial software package Abaqus with linear (C3D8H) and quadratic (C3D20H) elements and achieved an excellent agreement. To show the performance of our mixed formulation we used Schwarz Primitive mesh under compression with different element aspect ratio. In the case of thinner element (high aspect ratio), KSP iteration increased as expected but remains constant under mesh refinement, which shows the robustness of our mixed models. It should be noted that, the linear iteration count for schwarz mesh starts with large number due to higher aspect ratio of the element, but under mesh refinement we get smaller iteration count since elements become roughly isotropic for $l \geq 3$. We run all the benchmark problems in current configuration since the solve time is smaller than initial configuration as depicted for Schwarz mesh under compression. For all the experiment above, we can conclude that our mixed formulations are accurate and robust in both compressible in incompressible cases. Higher order element Q_3Q_1 (discontinuous pressure) and Q_3Q_{2c} (continuous pressure) are more accurate and we can reduce the cost of computation (KSP iteration) by choosing $\nu_p \geq 0$. Furthermore, we showed that for material with $\nu = 0.495$, despite the accuracy of the displacement result using single field formulation, its solve time is larger than mixed formulation. Therefore, we recommend using mixed formulation to simulate the nearly incompressible even if we only need displacement values.

Chapter 5

Conclusion and future work

We investigated various constitutive formulations for elasticity along with their implementation in finite element software packages. These formulations which are written based on the deformation gradient $\boldsymbol{F}$ are not numerically stable and lose several digits of accuracy. The instabilities in the standard formulations are due to the presence of function like $\log J$ which is ill-conditioned when $J \approx 1$ and the catastrophic cancellation that happens in $J - 1$ term or strain tensor written in terms of $\boldsymbol{F}$. We proposed equivalent stable formulations which are stable and have relative accuracy of $\epsilon_{\text{machine}}$. These new formulations make use of displacement gradient $\boldsymbol{H}$ to define strain tensor without loss of significance and compute $J - 1$ term in a stable way. We computed the relative accuracy of both standard and stable formulations of $J - 1$ and Ogden material model and exhibited the results up to strain of order 10^{-8}. With single precision in standard formulation we lose all the accuracy of the final results while using stable formulations give the accuracy of $\epsilon_{\text{machine}}$.

We considered the isochoric-volumetric split formulation and discretized it in a matrix-free way for single and mixed formulation with hyperelastic Neo-Hookean, Mooney-Rivlin and Ogden models. The displacement-based formulation exhibited non-physical results in pressure response despite using high-order basis function. However, mixed formulation was an an efficient approach to overcome locking and have a smooth pressure distribution as shown in punch test. To the knowledge of the author, most packages available for mixed problem, are using direct solver for the block

preconditioner. However, we implemented the upper triangular block preconditioner with p-MG on displacement block, which offers an efficient and robust convergence for different problem using structure and unstructured elements under different loading situations. Moreover, we showed that we can solve a mixed problem at similar cost to the best available compressible solver (single field). In nearly incompressibility limit, despite having larger system to solve using mixed formulation, the solve time is almost half of the single field model. Furthermore, we implemented all material models in current configuration, which requires less storage and flops in the tangent operator thus faster computation. It should be noted that we also implemented the perturbed Lagrange-multiplier mixed formulation where unlike non-symmetric formulation, we have an objective functional and one can use backtracking line search in nonlinear solver. With the mixed model implemented in Ratel, we can simulate all other mixed problems such as poroelasticity, phase field modeling, just by defining the operators for these models.

Bibliography

[1] A. Abdelfattah, H. Anzt, E. G. Boman, E. Carson, T. Cojean, J. Dongarra, A. Fox, M. Gates, N. J. Higham, X. S. Li, et al., A survey of numerical linear algebra methods utilizing mixed-precision arithmetic, The International Journal of High Performance Computing Applications, 35 (2021), pp. 344–369, https://doi.org/10.1177/10943420211003313.

[2] A. Abdelfattah, V. Barra, N. Beams, J. Brown, J.-S. Camier, V. Dobrev, Y. Dudouit, L. Ghaffari, S. Grimberg, T. Kolev, D. Medina, W. Pazner, T. Ratnayaka, R. Shakeri, J. L. Thompson, S. Tomov, and J. Wright III, libCEED user manual, Nov. 2023, https://doi.org/10.5281/zenodo.10062388.

[3] N. T. Aboulkhair, M. Simonelli, L. Parry, I. Ashcroft, C. Tuck, and R. Hague, 3d printing of aluminium alloys: Additive manufacturing of aluminium alloys using selective laser melting, Progress in Materials Science, 106 (2019), p. 100578.

[4] J. H. Adler, L. Dorfmann, D. Han, S. MacLachlan, and C. Paetsch, Mathematical and computational models of incompressible materials subject to shear, IMA Journal of Applied Mathematics, 79 (2014), pp. 889–914, https://doi.org/10.1093/imamat/hxu022.

[5] D. N. Arnold and M. E. Rognes, Stability of lagrange elements for the mixed laplacian, Calcolo, 46 (2009), pp. 245–260, https://doi.org/10.1007/s10092-009-0009-6.

[6] Z. Atkins, J. Brown, L. Ghaffari, R. Shakeri, R. Stengel, and J. L. Thompson, Ratel user manual, Zenodo, (2023), https://doi.org/10.5281/zenodo.10063890, 10.5281/zenodo.10063890.

[7] F. Auricchio, L. Beirão da Veiga, C. Lovadina, A. Reali, R. L. Taylor, and P. Wriggers, Approximation of incompressible large deformation elastic problems: some unresolved issues, Computational Mechanics, 52 (2013), pp. 1153–1167, https://doi.org/10.1007/s00466-013-0869-0.

[8] I. Babuška, The finite element method with lagrangian multipliers, Numerische Mathematik, 20 (1973), pp. 179–192.

[9] S. Balay, S. Abhyankar, M. F. Adams, S. Benson, J. Brown, P. Brune, K. Buschelman, E. Constantinescu, L. Dalcin, A. Dener, V. Eijkhout, J. Faibussowitsch,

W. D. Gropp, V. Hapla, T. Isaac, P. Jolivet, D. Karpeev, D. Kaushik, M. G. Knepley, F. Kong, S. Kruger, D. A. May, L. C. McInnes, R. T. Mills, L. Mitchell, T. Munson, J. E. Roman, K. Rupp, P. Sanan, J. Sarich, B. F. Smith, S. Zampini, H. Zhang, H. Zhang, and J. Zhang, PETSc/TAO users manual, Tech. Report ANL-21/39 - Revision 3.19, Argonne National Laboratory, 2023, https://doi.org/10.2172/1968587.

[10] K.-J. Bathe, Finite element procedures, Klaus-Jurgen Bathe, 2006.

[11] N. H. Beebe, Computation of expm1(x) = exp(x) − 1. Online: https://www.math.utah.edu/~beebe/reports/expm1.pdf, 2002.

[12] N. H. F. Beebe, The Mathematical-Function Computation Handbook - Programming Using the MathCW Portable Software Library, Springer, 2017, https://doi.org/10.1007/978-3-319-64110-2, 10.1007/978-3-319-64110-2.

[13] M. Benzi, Preconditioning techniques for large linear systems: A survey, Journal of Computational Physics, 182 (2002), pp. 418–477, https://doi.org/10.1006/jcph.2002.7176, https://www.sciencedirect.com/science/article/pii/S0021999102971767.

[14] L. Bertagna, S. Deparis, L. Formaggia, D. Forti, and A. Veneziani, The lifev library: engineering mathematics beyond the proof of concept, arXiv math.NA, (2017), https://doi.org/10.48550/arXiv.1710.06596, https://arxiv.org/abs/1710.06596.

[15] S. Bieber, F. Auricchio, A. Reali, and M. Bischoff, Artificial instabilities of finite elements for nonlinear elasticity: Analysis and remedies, International Journal for Numerical Methods in Engineering, 124 (2023), pp. 2638–2675, https://doi.org/10.1002/nme.7224, https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.7224, https://arxiv.org/abs/https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.7224.

[16] Å. Björck et al., Numerical methods in matrix computations, vol. 59, Springer, 2015, https://doi.org/10.1007/978-3-319-05089-8.

[17] C. Böhm, J. Korelc, B. Hudobivnik, A. Kraus, and P. Wriggers, Mixed virtual element formulations for incompressible and inextensible problems, Computational Mechanics, (2023), pp. 1–34, https://doi.org/10.1007/s00466-023-02340-9.

[18] J. Bonet and R. D. Wood, Nonlinear Continuum Mechanics for Finite Element Analysis, Cambridge University Press, 2 ed., 2008, https://doi.org/10.1017/CBO9780511755446.

[19] S. C. Brenner, The mathematical theory of finite element methods, Springer, 2008, https://doi.org/10.1007/978-0-387-75934-0.

[20] F. Brezzi, On the existence, uniqueness and approximation of saddle-point problems arising from lagrangian multipliers, Publications mathématiques et informatique de Rennes, (1974), pp. 1–26.

[21] F. Brezzi and M. Fortin, Mixed and hybrid finite element methods, vol. 15, Springer Science & Business Media, 2012.

[22] U. Brink and E. Stein, On some mixed finite element methods for incompressible and nearly incompressible finite elasticity, Computational Mechanics, 19 (1996), pp. 105–119, https://doi.org/10.1007/BF02824849.

[23] A. N. Brooks and T. J. Hughes, Streamline upwind/petrov-galerkin formulations for convection dominated flows with particular emphasis on the incompressible navier-stokes equations, Computer Methods in Applied Mechanics and Engineering, 32 (1982), pp. 199–259, https://doi.org/10.1016/0045-7825(82)90071-8, https://www.sciencedirect.com/science/article/pii/0045782582900718.

[24] J. Brown, Efficient Nonlinear Solvers for Nodal High-Order Finite Elements in 3D, Journal of Scientific Computing, 45 (2010), https://doi.org/10.1007/s10915-010-9396-8.

[25] J. Brown, A. Abdelfattah, V. Barra, N. Beams, J.-S. Camier, V. Dobrev, Y. Dudouit, L. Ghaffari, T. Kolev, D. Medina, et al., libceed: Fast algebra for high-order element-based discretizations, Journal of Open Source Software, 6 (2021), p. 2945.

[26] J. Brown, V. Barra, N. Beams, L. Ghaffari, M. Knepley, W. Moses, R. Shakeri, K. Stengel, J. L. Thompson, and J. Zhang, Performance portable solid mechanics via matrix-free p-multigrid, 2022, https://arxiv.org/abs/2204.01722.

[27] D. Chapelle and K. Bathe, The inf-sup test, Computers & Structures, 47 (1993), pp. 537–545, https://doi.org/10.1016/0045-7949(93)90340-J, https://www.sciencedirect.com/science/article/pii/004579499390340J.

[28] S. J. Connolly, D. Mackenzie, and Y. Gorash, Isotropic hyperelasticity in principal stretches: explicit elasticity tensors and numerical implementation, Computational Mechanics, 64 (2019), pp. 1273–1288, https://doi.org/10.1007/s00466-019-01707-1.

[29] D. Davydov, J.-P. Pelteret, D. Arndt, M. Kronbichler, and P. Steinmann, A matrix-free approach for finite-strain hyperelastic problems using geometric multigrid, International Journal for Numerical Methods in Engineering, 121 (2020), pp. 2874–2895, https://doi.org/10.1002/nme.6336, https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.6336, https://arxiv.org/abs/https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.6336.

[30] E. de Souza Neto, D. Perić, G. Huang, and D. Owen, Remarks on the stability of enhanced strain elements in finite elasticity and elastoplasticity, Communications in Numerical Methods in Engineering, 11 (1995), pp. 951–961.

[31] L. Demkowicz, J. T. Oden, W. Rachowicz, and O. Hardy, Toward a universal hp adaptive finite element strategy, part 1. constrained approximation and data structure, Computer Methods in Applied Mechanics and Engineering, 77 (1989), pp. 79–112.

[32] S. Doll and K. Schweizerhof, On the development of volumetric strain energy functions, Journal of Applied Mechanics, 67 (2000), pp. 17–21, https://doi.org/10.1115/1.321146, https://api.semanticscholar.org/CorpusID:120833701.

[33] T. Elguedj, Y. Bazilevs, V. Calo, and T. Hughes, $\bar{B}$ and $\bar{f}$ projection methods for nearly incompressible linear and non-linear elasticity and plasticity using higher-order nurbs elements, Computer Methods in Applied Mechanics and Engineering, 197 (2008), pp. 2732–2762, https://doi.org/10.1016/j.cma.2008.01.012, https://www.sciencedirect.com/science/article/pii/S0045782508000248.

[34] H. C. Elman, D. J. Silvester, and A. J. Wathen, Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics, Oxford University Press, USA, 2014.

[35] P. E. Farrell, L. Mitchell, L. R. Scott, and F. Wechsung, Robust multigrid methods for nearly incompressible elasticity using macro elements, IMA Journal of Numerical Analysis, 42 (2022), pp. 3306–3329, https://doi.org/10.1093/imanum/drab083.

[36] N. Fehn, P. Munch, W. A. Wall, and M. Kronbichler, Hybrid multigrid methods for high-order discontinuous galerkin discretizations, Journal of Computational Physics, 415 (2020), p. 109538, https://doi.org/10.1016/j.jcp.2020.109538, https://www.sciencedirect.com/science/article/pii/S0021999120303120.

[37] P. Fischer, M. Min, T. Rathnayake, S. Dutta, T. Kolev, V. Dobrev, J.-S. Camier, M. Kronbichler, T. Warburton, K. Swirydowicz, and J. Brown, Scalability of high-performance PDE solvers, International Journal of High Performance Computing Applications, 34 (2020), https://doi.org/10.1177%2F1094342020915762, https://www.osti.gov/biblio/1660236.

[38] A. J. Gil, C. H. Lee, J. Bonet, and M. Aguirre, A stabilised petrov–galerkin formulation for linear tetrahedral elements in compressible, nearly incompressible and truly incompressible fast dynamics, Computer Methods in Applied Mechanics and Engineering, 276 (2014), pp. 659–690, https://doi.org/10.1016/j.cma.2014.04.006, https://www.sciencedirect.com/science/article/pii/S0045782514001315.

[39] S. Govindjee. Personal communication, 2022.

[40] O. Gültekin, H. Dal, and G. A. Holzapfel, On the quasi-incompressible finite element analysis of anisotropic hyperelastic materials, Computational mechanics, 63 (2019), pp. 443–453, https://doi.org/10.1007/s00466-018-1602-9.

[41] V. Gurev, P. Pathmanathan, J.-L. Fattebert, H.-F. Wen, J. Magerlein, R. A. Gray, D. F. Richards, and J. J. Rice, A high-resolution computational model of the deforming human heart, Biomechanics and modeling in mechanobiology, 14 (2015), pp. 829–849, https://doi.org/10.1007/s10237-014-0639-8.

[42] I. Harari and U. Albocher, Computation of eigenvalues of a real, symmetric 3× 3 matrix with particular reference to the pernicious case of two nearly equal eigenvalues, International Journal for Numerical Methods in Engineering, 124 (2023), pp. 1089–1110, https://doi.org/10.1002/nme.7153.

[43] M. R. Hestenes and E. Stiefel, Methods of conjugate gradients for solving linear systems, vol. 49, NBS Washington, DC, 1952.

[44] G. Holzapfel, Nonlinear solid mechanics: a continuum approach for engineering, Wiley, Chichester New York, 2000.

[45] T. J. Hughes, The finite element method: linear static and dynamic finite element analysis, Courier Corporation, 2012.

[46] T. J. Hughes, L. P. Franca, and M. Balestra, A new finite element formulation for computational fluid dynamics: V. circumventing the babuška-brezzi condition: a stable petrov-galerkin formulation of the stokes problem accommodating equal-order interpolations, Computer Methods in Applied Mechanics and Engineering, 59 (1986), pp. 85–99, https://doi.org/10.1016/0045-7825(86)90025-3, https://www.sciencedirect.com/science/article/pii/0045782586900253.

[47] T. J. Hughes, L. P. Franca, and G. M. Hulbert, A new finite element formulation for computational fluid dynamics: Viii. the galerkin/least-squares method for advective-diffusive equations, Computer Methods in Applied Mechanics and Engineering, 73 (1989), pp. 173–189, https://doi.org/10.1016/0045-7825(89)90111-4, https://www.sciencedirect.com/science/article/pii/0045782589901114.

[48] T. J. R. Hughes, Generalization of selective integration procedures to anisotropic and nonlinear media, Int. J. Numer. Methods Eng., 15 (1980), pp. 1413–1418, https://doi.org/10.1002/nme.1620150914.

[49] M. Innes, Don't unroll adjoint: Differentiating ssa-form programs, CoRR, (2018), https://doi.org/10.48550/arXiv.1810.07951, http://arxiv.org/abs/1810.07951.

[50] I. C. F. Ipsen, A note on preconditioning nonsymmetric matrices, SIAM Journal on Scientific Computing, 23 (2001), pp. 1050–1051, https://doi.org/10.1137/S1064827500377435, 10.1137/S1064827500377435, https://arxiv.org/abs/10.1137/S1064827500377435.

[51] Ł. Kaczmarczyk, Z. Ullah, K. Lewandowski, X. Meng, X.-Y. Zhou, I. Athanasiadis, H. Nguyen, C.-A. Chalons-Mouriesse, E. J. Richardson, E. Miur, A. G. Shvarts, M. Wakeni, and C. J. Pearce, Mofem: An open source, parallel finite element library, Journal of Open Source Software, 5 (2020), p. 1441, https://doi.org/10.21105/joss.01441.

[52] C. Kadapa, W. Dettmer, and D. Perić, Subdivision based mixed methods for isogeometric analysis of linear and nonlinear nearly incompressible materials, Computer Methods in Applied Mechanics and Engineering, 305 (2016), pp. 241–270, https://doi.org/10.1016/j.cma.2016.03.013, https://www.sciencedirect.com/science/article/pii/S0045782516300950.

[53] C. Kadapa and M. Hossain, A linearized consistent mixed displacement-pressure formulation for hyperelasticity, Mechanics of Advanced Materials and Structures, 29 (2022), pp. 267–284.

[54] S. C. Kapfer, S. T. Hyde, K. Mecke, C. H. Arns, and G. E. Schröder-Turk, Minimal surface scaffold designs for tissue engineering, Biomaterials, 32 (2011), pp. 6875–6882, https://doi.org/10.1016/j.biomaterials.2011.06.012.

[55] E. Karabelas, M. A. Gsell, G. Haase, G. Plank, and C. M. Augustin, An accurate, robust, and efficient finite element framework with applications to anisotropic, nearly and fully incompressible elasticity, Computer Methods in Applied Mechanics and Engineering, 394 (2022), p. 114887, https://doi.org/10.1016/j.cma.2022.114887, https://www.sciencedirect.com/science/article/pii/S0045782522001803.

[56] O. Klaas, A. Maniatty, and M. S. Shephard, A stabilized mixed finite element method for finite elasticity.: Formulation for linear displacement and pressure interpolation, Computer Methods in Applied Mechanics and Engineering, 180 (1999), pp. 65–79, https://doi.org/10.1016/S0045-7825(99)00059-6, https://www.sciencedirect.com/science/article/pii/S0045782599000596.

[57] M. Kronbichler and K. Ljungkvist, Multigrid for matrix-free high-order finite element computations on graphics processors, ACM Transactions on Parallel Computing (TOPC), 6 (2019), pp. 1–32, https://doi.org/10.1145/3322813.

[58] A. D. Lindsay, D. R. Gaston, C. J. Permann, J. M. Miller, D. Andrš, A. E. Slaughter, F. Kong, J. Hansel, R. W. Carlsen, C. Icenhour, L. Harbour, G. L. Giudicelli, R. H. Stogner, P. German, J. Badger, S. Biswas, L. Chapuis, C. Green, J. Hales, T. Hu, W. Jiang, Y. S. Jung, C. Matthews, Y. Miao, A. Novak, J. W. Peterson, Z. M. Prince, A. Rovinelli, S. Schunert, D. Schwen, B. W. Spencer, S. Veeraraghavan, A. Recuero, D. Yushu, Y. Wang, A. Wilkins, and C. Wong, 2.0 - MOOSE: Enabling massively parallel multiphysics simulation, SoftwareX, 20 (2022), p. 101202, https://doi.org/10.1016/j.softx.2022.101202, https://www.sciencedirect.com/science/article/pii/S2352711022001200.

[59] A. Logg, K.-A. Mardal, and G. Wells, Automated solution of differential equations by the finite element method: The FEniCS book, vol. 84, Springer Science & Business Media, 2012, https://doi.org/10.1007/978-3-642-23099-8.

[60] S. A. Maas, B. J. Ellis, G. A. Ateshian, and J. A. Weiss, FEBio: Finite elements for biomechanics, Journal of Biomechanical Engineering, 134 (2012), p. 011005, https://doi.org/10.1115/1.4005694, 10.1115/1.4005694, https://arxiv.org/abs/https://asmedigitalcollection.asme.org/biomechanical/article-pdf/134/1/011005/5665064/011005_1.pdf.

[61] D. S. Malkus and T. J. Hughes, Mixed finite element methods — reduced and selective integration techniques: A unification of concepts, Computer Methods in Applied Mechanics and Engineering, 15 (1978), pp. 63–81, https://doi.org/10.1016/0045-7825(78)90005-1, https://www.sciencedirect.com/science/article/pii/0045782578900051.

[62] A. M. Maniatty, Y. Liu, O. Klaas, and M. S. Shephard, Higher order stabilized finite element method for hyperelastic finite deformation, Computer Methods in Applied Mechanics

and Engineering, 191 (2002), pp. 1491–1503, https://doi.org/10.1016/S0045-7825(01)00335-8, https://www.sciencedirect.com/science/article/pii/S0045782501003358.

[63] I. Maskery, L. Parry, D. Padrão, R. Hague, and I. Ashcroft, Flatt pack: A research-focussed lattice design program, Additive Manufacturing, 49 (2022), p. 102510, https://doi.org/10.1016/j.addma.2021.102510.

[64] I. Maskery, L. Sturm, A. O. Aremu, A. Panesar, C. B. Williams, C. J. Tuck, R. D. Wildman, I. A. Ashcroft, and R. J. Hague, Insights into the mechanical properties of several triply periodic minimal surface lattice structures made by polymer additive manufacturing, Polymer, 152 (2018), pp. 62–71, https://doi.org/10.1016/j.polymer.2017.11.049.

[65] W. McLean and W. C. H. McLean, Strongly elliptic systems and boundary integral equations, Cambridge university press, 2000.

[66] A. Mehraban, J. Brown, H. Tufo, J. Thompson, R. Shakeri, and R. Regueiro, Efficient parallel scalable matrix-free 3d high-order finite element simulation of neo-hookean compressible hyperelasticity at finite strain, Volume 12: Mechanics of Solids, Structures, and Fluids; Micro- and Nano- Systems Engineering and Packaging (2021), p. V012T12A027, https://doi.org/10.1115/IMECE2021-70768, 10.1115/IMECE2021-70768, https://arxiv.org/abs/https://asmedigitalcollection.asme.org/IMECE/proceedings-pdf/IMECE2021/85680/V012T12A027/6829164/v012t12a027-imece2021-70768.pdf.

[67] C. Miehe, Aspects of the formulation and finite element implementation of large strain isotropic elasticity, International Journal for Numerical Methods in Engineering, 37 (1994), pp. 1981–2004, https://doi.org/10.1002/nme.1620371202, https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.1620371202, https://arxiv.org/abs/https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.1620371202.

[68] K. M. Moerman, B. Fereidoonnezhad, and J. P. McGarry, Novel hyperelastic models for large volumetric deformations, International Journal of Solids and Structures, 193-194 (2020), pp. 474–491, https://doi.org/10.1016/j.ijsolstr.2020.01.019, https://www.sciencedirect.com/science/article/pii/S0020768320300251.

[69] M. F. Murphy, G. H. Golub, and A. J. Wathen, A note on preconditioning for indefinite linear systems, SIAM Journal on Scientific Computing, 21 (2000), pp. 1969–1972, https://doi.org/10.1137/S1064827599355153, 10.1137/S1064827599355153, https://arxiv.org/abs/10.1137/S1064827599355153.

[70] E. D. S. Neto, F. A. Pires, and D. Owen, F-bar-based linear triangles and tetrahedra for finite strain analysis of nearly incompressible solids. part i: formulation and benchmarking, International Journal for Numerical Methods in Engineering, 62 (2005), pp. 353–383.

[71] J. T. Oden, L. Demkowicz, W. Rachowicz, and T. A. Westermann, Toward a universal hp adaptive finite element strategy, part 2. a posteriori error estimation, Computer methods in applied mechanics and engineering, 77 (1989), pp. 113–180.

[72] D. Pantuso and K.-J. Bathe, On the stability of mixed finite elements in large strain analysis of incompressible solids, Finite Elements in Analysis and Design, 28 (1997), pp. 83–104, https://doi.org/10.1016/S0168-874X(97)81953-1, https://www.sciencedirect.com/science/article/pii/S0168874X97819531.

[73] P. J. Phillips and M. F. Wheeler, Overcoming the problem of locking in linear elasticity and poroelasticity: an heuristic approach, Computational Geosciences, 13 (2009), pp. 5–12, https://doi.org/10.1007/s10596-008-9114-x.

[74] W. Rachowicz, J. T. Oden, and L. Demkowicz, Toward a universal hp adaptive finite element strategy part 3. design of hp meshes, Computer Methods in Applied Mechanics and Engineering, 77 (1989), pp. 181–212.

[75] S. Reese and P. Wriggers, A stabilization technique to avoid hourglassing in finite elasticity, International Journal for Numerical Methods in Engineering, 48 (2000), pp. 79–109, https://doi.org/10.1002/(SICI)1097-0207(20000510)48, https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291097-0207%2820000510%2948%3A1%3C79%3A%3AAID-NME869%3E3.0.CO%3B2-D, https://arxiv.org/abs/https://onlinelibrary.wiley.com/doi/pdf/10.1002/%28SICI%291097-0207%2820000510%2948%3A1%3C79%3A%3AAID-NME869%3E3.0.CO%3B2-D.

[76] R. A. Regueiro, Cven 7511: Computational finite inelasticity and multiphase mechanics. university of colorado boulder, Sept. 2020, https://experts.colorado.edu/display/coursename_CVEN-7511.

[77] E. M. Rønquist and A. T. Patera, Spectral element multigrid. I. formulation and numerical results, Journal of Scientific Computing, 2 (1987), pp. 389–406.

[78] Y. Saad, Iterative methods for sparse linear systems, SIAM, 2003, https://doi.org/10.1137/1.9780898718003.

[79] A. G. Salinger, R. A. Bartlett, A. M. Bradley, Q. Chen, I. P. Demeshko, X. Gao, G. A. Hansen, A. Mota, R. P. Muller, E. Nielsen, et al., Albany: using component-based design to develop a flexible, generic multiphysics analysis code, International Journal for Multiscale Computational Engineering, 14 (2016), https://doi.org/10.1615/IntJMultCompEng.2016017040.

[80] R. Shakeri, Linear and mixed-linear 2d mms elasticity with assembled and matrix-free approaches, Aug. 2023, https://doi.org/10.5281/zenodo.8287424, 10.5281/zenodo.8287424.

[81] R. Shakeri, J. Brown, L. Ghaffari, and J. Thompson, Demo code for stable numerics, Zenodo, (2024), https://doi.org/10.5281/zenodo.10553116.

[82] J. R. Shewchuk, An introduction to the conjugate gradient method without the agonizing pain, 1994.

[83] Y.-h. Shih, G. Stadler, and F. Wechsung, Robust multigrid techniques for augmented lagrangian preconditioning of incompressible stokes equations with extreme viscosity variations, SIAM Journal on Scientific Computing, (2022), pp. S27–S53, https://doi.org/10.1137/21M1430698.

[84] J. C. Simo and M. Rifai, A class of mixed assumed strain methods and the method of incompatible modes, International journal for numerical methods in engineering, 29 (1990), pp. 1595–1638.

[85] M. Smith, ABAQUS/Standard User's Manual, Version 6.9, Dassault Systèmes Simulia Corp, United States, 2009.

[86] R. Srinivasan and R. Perucchio, Finite element analysis of anisotropic non-linear incompressible elastic solids by a mixed model, International Journal for Numerical Methods in Engineering, 37 (1994), pp. 3075–3092, https://doi.org/10.1002/nme.1620371805, https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.1620371805, https://arxiv.org/abs/https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.1620371805.

[87] T. Sussman and K.-J. Bathe, A finite element formulation for nonlinear incompressible elastic and inelastic analysis, Computers & Structures, 26 (1987), pp. 357–409, https://doi.org/10.1016/0045-7949(87)90265-3, https://www.sciencedirect.com/science/article/pii/0045794987902653.

[88] R. L. Taylor, FEAP - finite element analysis program. Online: http://projects.ce.berkeley.edu/feap/, 2014.

[89] A. Ten Eyck and A. Lew, An adaptive stabilization strategy for enhanced strain methods in non-linear elasticity, International Journal for Numerical Methods in Engineering, 81 (2010), pp. 1387–1416, https://doi.org/10.1002/nme.2734, https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.2734, https://arxiv.org/abs/https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.2734.

[90] L. N. Trefethen and D. Bau, III, Numerical Linear Algebra, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1997, https://doi.org/10.1137/1.9780898719574.

[91] W. A. Wall, M. Bischoff, and E. Ramm, A deformation dependent stabilization technique, exemplified by eas elements at large strains, Computer Methods in Applied Mechanics and Engineering, 188 (2000), pp. 859–871, https://doi.org/10.1016/S0045-7825(99)00365-5, https://www.sciencedirect.com/science/article/pii/S0045782599003655. IVth World Congress on Computational Mechanics. (II). Optimum.

[92] P. Wriggers, Nonlinear finite element methods, Springer Science & Business Media, 2008, https://doi.org/10.1007/978-3-540-71001-1.

[93] P. Wriggers, M. De Bellis, and B. Hudobivnik, A taylor–hood type virtual element formulations for large incompressible strains, Computer Methods in Applied Mechanics and Engineering, 385 (2021), p. 114021, https://doi.org/10.1016/j.cma.2021.114021, https://www.sciencedirect.com/science/article/pii/S0045782521003522.

[94] P. Wriggers, B. D. Reddy, W. Rust, and B. Hudobivnik, Efficient virtual element formulations for compressible and incompressible finite deformations, Computational Mechanics, 60 (2017), pp. 253–268, https://doi.org/10.1007/s00466-017-1405-4.

[95] P. Wriggers and S. Reese, A note on enhanced strain methods for large deformations, Computer Methods in Applied Mechanics and Engineering, 135 (1996), pp. 201–209, https://doi.org/10.1016/0045-7825(96)01037-7, https://www.sciencedirect.com/science/article/pii/0045782596010377.

Appendix A

Preliminaries

The mechanical response of a deformable body can be described by a motion $\phi(\boldsymbol{X}, t)$ which will generally change its shape, position and orientation. According to Figure A.1 the material point $\boldsymbol{X} \in \Omega_0$ at $t = 0$ (initial configuration) is transformed onto spatial point $\boldsymbol{x} = \phi(\boldsymbol{X}, t) \in \Omega$ at time t (current configuration). The deformation gradient [44]

$$\boldsymbol{F} \coloneqq \nabla_X \phi(\boldsymbol{X}, t) = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{X}} = \boldsymbol{I} + \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{X}} \tag{A.1}$$

maps the unit tangent $d\boldsymbol{X}$ of the initial/Lagrangian configuration onto its counterpart $d\boldsymbol{x}$ in the current/Eulerian configuration, where we used the definition of displacement vector

$$\boldsymbol{u} = \boldsymbol{x} - \boldsymbol{X} \tag{A.2}$$

Moreover, we define the change in volume between the two configurations by

$$dv = J(\boldsymbol{X}, t) dV, \quad J = \det \boldsymbol{F} > 0 \tag{A.3}$$

where dV and dv denote infinitesimal volume elements defined in the initial and current configurations. The vector elements of infinitesimally small areas $d\boldsymbol{a}$ and $d\boldsymbol{A}$ defined in the current and initial configurations are related through [44]

$$d\boldsymbol{a} = J \boldsymbol{F}^{-T} d\boldsymbol{A} \tag{A.4}$$

which is well-known as Nanson's formula.

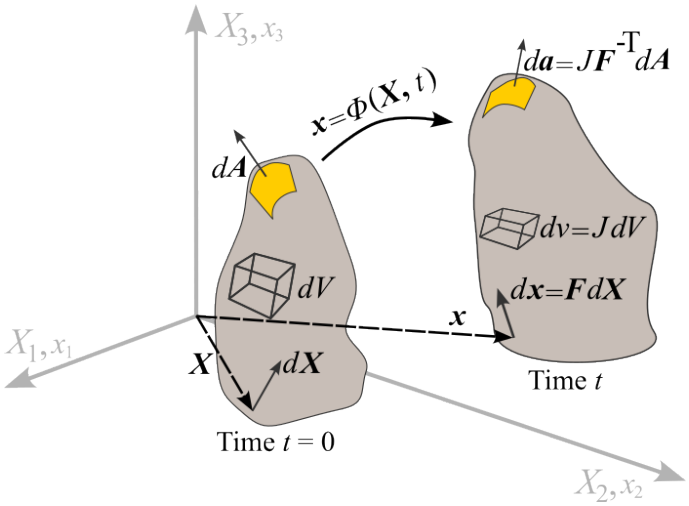


Figure A.1: General motion of a deformable body

Under the motion ϕ, the principal stretches $\lambda_1, \lambda_2, \lambda_3$ of this deformation are the eigenvalues of unique SPD right (or left) stretch tensor $\mathbf{U}$ (or $\mathbf{v}$)

$$\mathbf{U}\boldsymbol{N}_i = \lambda_i \boldsymbol{N}_i, \quad \text{or} \quad \mathbf{v}\boldsymbol{n}_i = \lambda_i \boldsymbol{n}_i, \quad i = 1, 2, 3 \tag{A.5}$$

where the eigenvectors are related through $\boldsymbol{n}_i = \boldsymbol{R}\boldsymbol{N}_i$ and stretch tensors are defined by the polar decomposition of the deformation gradient $\boldsymbol{F} = \boldsymbol{R}\mathbf{U} = \mathbf{v}\boldsymbol{R}$ with $\boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{I}$ as a rotation tensor. In initial (Time=0) and current (Time=t) configurations we define the right and left Cauchy–Green tensors by

$$\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F} = \mathbf{U}^2. \tag{A.6}$$

$$\boldsymbol{b} = \boldsymbol{F}\boldsymbol{F}^T = \mathbf{v}^2. \tag{A.7}$$

and Green-Lagrange and Green-Euler strains

$$\boldsymbol{E} = \frac{1}{2}\left(\boldsymbol{C} - \boldsymbol{I}\right) = \frac{1}{2}\left(\boldsymbol{H} + \boldsymbol{H}^T + \boldsymbol{H}^T\boldsymbol{H}\right). \tag{A.8}$$

$$\boldsymbol{e} = \frac{1}{2}\left(\boldsymbol{b} - \boldsymbol{I}\right) = \frac{1}{2}\left(\boldsymbol{H} + \boldsymbol{H}^T + \boldsymbol{H}\boldsymbol{H}^T\right). \tag{A.9}$$

in which $\boldsymbol{H} = \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{X}}$. Clearly $\mathbf{U}$, $\boldsymbol{C}$ and $\boldsymbol{E}$ have the **same** orthonormal eigenvectors $\boldsymbol{N}_i$ and their

eigenvalues are related through $\lambda_i^C = \lambda_i^2 = 1 + 2\lambda_i^E$ i.e.,

$$\begin{aligned} \boldsymbol{C}\boldsymbol{N}_i &= \lambda_i^C \boldsymbol{N}_i = \lambda_i^2 \boldsymbol{N}_i. \quad i = 1, 2, 3 \\ \boldsymbol{E}\boldsymbol{N}_i &= \lambda_i^E \boldsymbol{N}_i = \frac{1}{2}\left(\lambda_i^2 - 1\right)\boldsymbol{N}_i. \quad i = 1, 2, 3, \end{aligned} \tag{A.10}$$

Similarly the eigenvectors of $\mathbf{v}$, $\boldsymbol{b}$ and $\boldsymbol{e}$ are $\boldsymbol{n}_i$ and their eigenvalues satisfy $\lambda_i^b = \lambda_i^2 = 1 + 2\lambda_i^e$ i.e.,

$$\begin{aligned} \boldsymbol{b}\boldsymbol{n}_i &= \lambda_i^b \boldsymbol{n}_i = \lambda_i^2 \boldsymbol{n}_i. \quad i = 1, 2, 3 \\ \boldsymbol{e}\boldsymbol{n}_i &= \lambda_i^e \boldsymbol{n}_i = \frac{1}{2}\left(\lambda_i^2 - 1\right)\boldsymbol{n}_i. \quad i = 1, 2, 3, \end{aligned} \tag{A.11}$$

Appendix B

Numerical Stability Evaluation

In order to compare stability of different implementations of functions of the displacement gradient $\boldsymbol{H} = \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{X}}$, we start with by sampling $\boldsymbol{H}$ as the absolute value of a standard normal distribution (`H = abs.(randn(3, 3))` in Julia) and then plot relative error of each function $f(\epsilon \boldsymbol{H})$ as in Listing B.1, where $\epsilon \in (10^{-8}, 10^{-1})$ to cover a range from small to large strain. Julia's `big` converts the input to arbitrary precision (default gives $\epsilon_{\text{machine}} < 10^{-77}$) and further operations retain that arbitrary precision. For the range of ϵ considered, the `big` arithmetic can be considered exact. reference and then calculate the relative error for single and double precision i.e., `repr = Float32` and `repr = Float64`.

Listing B.1: Julia code for computing relative error of function $f(\epsilon \boldsymbol{H})$

```
function rel_error(eps, f, repr)
    ref = f(big.(eps*H)) # arbitrary precision
    norm(f(repr.(eps*H)) - ref) / norm(ref)
end
```

We start with the $J-1$ term which appears in all hyperelastic models and compare it with its stable form `J_m1` (2.10) as defined in Julia in Listing B.2 and their relative errors are shown in figure (2.2). As expected, the stable computation `J_m1` has a relative accuracy of order $\epsilon_{\text{machine}}$ for single and double precision and $J-1$ loses accuracy as $\boldsymbol{H}$ decreases. In fact, for $u_{i,j}$ of order 10^{-8} we can

trust no digits in single precision and we lost half of the digits in double precision, respectively.

Listing B.2: Code for computing $J - 1$ and `J_1`

```
function Jm1_unstable(H)
    F = I + H
    J = det(F)
    J - 1
end

function Jm1_stable(H)
    det_H = det(H)
    A1 = H[1,1]*H[2,2] + H[1,1]*H[3,3] + H[2,2]*H[3,3]
    A2 = H[1,2]*H[2,1] + H[1,3]*H[3,1] + H[2,3]*H[3,2]
    # Compute J-1
    det_H + tr(H) + A1 - A2
end
```

To assess stability of constitutive models, we implement the standard and unstable expressions for the appropriate stress $\boldsymbol{S}(\boldsymbol{H})$ or $\boldsymbol{\tau}(\boldsymbol{H})$, internally making use of Green-Lagrange or Green-Euler strains computed by stable means (2.3), and measure the relative error via Listing B.1, yielding figures like Figure 2.3 and Figure 2.5.

B.1 Accurate evaluation of `log1p_minus_x`

We require efficient and accurate evaluation of the function `log1p_minus_x`$(x) = \log(1 + x) - x$ to ensure accuracy of the AD-computed stresses in section 2.3. Figure B.1 demonstrates that $n = 6$ in (2.58) is sufficient to provide $O(\epsilon_{\text{machine}})$ accuracy evaluating `log1p_minus_x`(x) for $\epsilon \in (10^{-5}, 10^{-1})$. We believe it would be fruitful to develop a uniformly accurate algorithm for `log1p_minus_x` and include it in numerical libraries.

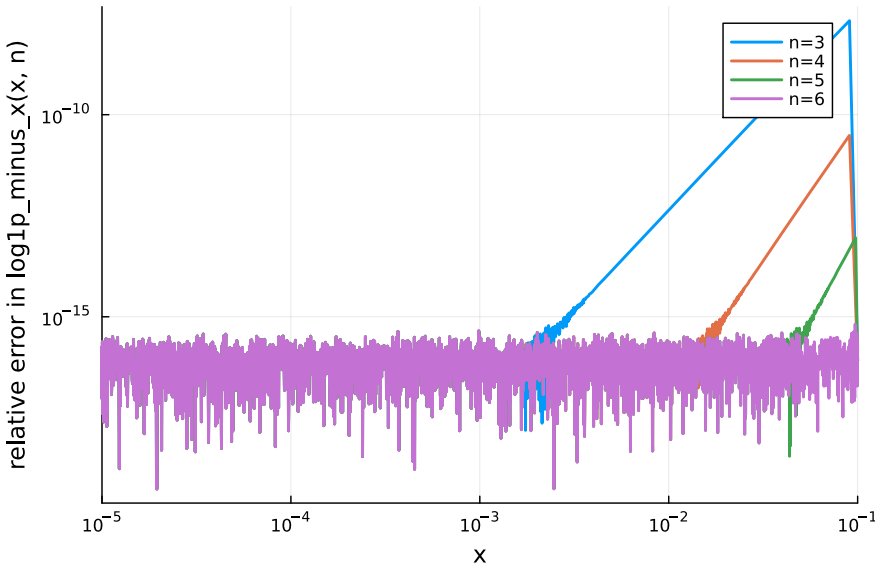


Figure B.1: Relative error of computation of `log1p_minus_x`() for different expansions of the Taylor series.

Appendix C

Stress linearization in initial configuration

We derived two forms of stress in section 3.2 and section 3.3 as

$$\boldsymbol{S} = \boldsymbol{S}_{\text{iso}} + \underbrace{\left(k_p JU' - pJ\right)\boldsymbol{C}^{-1}}_{\boldsymbol{S}_{\text{vol}}}, \tag{C.1}$$

$$\boldsymbol{S} = \boldsymbol{S}_{\text{iso}} + \underbrace{\left(k_p \hat{U} - p\right) J\hat{U}'\boldsymbol{C}^{-1}}_{\boldsymbol{S}_{\text{vol}}}, \tag{C.2}$$

where volumetric part is the same for all materials and the isochoric part $\boldsymbol{S}_{\text{iso}}$, for different materials is given in chapter 2.

The linearization of the volumetric form for both methods are given in (3.37) and (3.54). The isochoric linearization $\mathrm{d}\boldsymbol{S}_{\text{iso}}$ for Neo-Hookean is given in (3.36) and for Mooney-Rivlin and Ogden are as follow.

C.1 Mooney-Rivlin model

For the Mooney-Rivlin model (2.37), we have

$$\begin{aligned}
\mathrm{d}\boldsymbol{S}_{\mathrm{iso}} = &-\frac{4}{3}(\boldsymbol{C}^{-1}:\mathrm{d}\boldsymbol{E})\left(\mu_1 J^{-2/3} + 4\mu_2 J^{-4/3}\right)\boldsymbol{C}^{-1}\boldsymbol{E}_{\mathrm{dev}} \\
&- \frac{1}{3}\left(\mu_1 J^{-2/3} + 2\mu_2 J^{-4/3}\right)\left(2\mathbb{I}_1(\mathrm{d}\boldsymbol{E})\,\boldsymbol{C}^{-1} + \mathbb{I}_1\,\mathrm{d}\boldsymbol{C}^{-1}\right) \\
&- \frac{8}{3}(\boldsymbol{C}^{-1}:\mathrm{d}\boldsymbol{E})\mu_2 J^{-4/3}\left(\mathbb{I}_1(\boldsymbol{E})\boldsymbol{I}_3 - \boldsymbol{E}\right) \\
&+ 2\mu_2 J^{-4/3}\left(\mathbb{I}_1(\mathrm{d}\boldsymbol{E})\boldsymbol{I}_3 - \mathrm{d}\boldsymbol{E}\right) \\
&+ (c_1 + c_2)\boldsymbol{C}^{-1} - \frac{4}{3}\mu_2 J^{-4/3}\left(\mathbb{I}_1(\boldsymbol{E}) + 2\mathbb{I}_2(\boldsymbol{E})\right)\mathrm{d}\boldsymbol{C}^{-1}
\end{aligned} \tag{C.3}$$

where

$$\begin{aligned}
c_1 &= \frac{16}{9}\mu_2 J^{-4/3}(\boldsymbol{C}^{-1}:\mathrm{d}\boldsymbol{E})\left(\mathbb{I}_1(\boldsymbol{E}) + 2\mathbb{I}_2(\boldsymbol{E})\right). \\
c_2 &= -\frac{4}{3}\mu_2 J^{-4/3}\left(\mathbb{I}_1(\mathrm{d}\boldsymbol{E}) + 2\mathbb{I}_1(\boldsymbol{E})\mathbb{I}_1(\mathrm{d}\boldsymbol{E}) - 2\boldsymbol{E}:\mathrm{d}\boldsymbol{E}\right).
\end{aligned} \tag{C.4}$$

C.2 Ogden model

The linearization of Ogden model (2.43) is

$$\mathrm{d}\boldsymbol{S}_{\mathrm{iso}} = \sum_{i=1}^{3} \mathrm{d}s_i^{\mathrm{iso}}\boldsymbol{N}_i\boldsymbol{N}_i^T + s_i^{\mathrm{iso}}\left(\mathrm{d}\boldsymbol{N}_i\boldsymbol{N}_i^T + \boldsymbol{N}_i\,\mathrm{d}\boldsymbol{N}_i^T\right) \tag{C.5}$$

For example computing $\mathrm{d}s_1^{\mathrm{iso}}$ from (2.47) gives

$$\begin{aligned}
\mathrm{d}s_1^{\mathrm{iso}} = &-\frac{2\,\mathrm{d}\lambda_1}{\lambda_1^3}\sum_{j=1}^{N}\frac{m_j}{3}\left[2\,\mathtt{expm1}(\alpha_j\ell_1) - \mathtt{expm1}(\alpha_j\ell_2) - \mathtt{expm1}(\alpha_j\ell_3)\right]J^{-\alpha_j/3} \\
&+ \frac{1}{\lambda_1^2}\sum_{j=1}^{N}\frac{m_j\alpha_j}{3}\left[2\,\mathrm{d}\ell_1\exp(\alpha_j\ell_1) - \mathrm{d}\ell_2\exp(\alpha_j\ell_2) - \mathrm{d}\ell_3\exp(\alpha_j\ell_3)\right]J^{-\alpha_j/3} \\
&- \frac{1}{\lambda_1^2}\sum_{j=1}^{N}\frac{m_j\alpha_j}{9}\left[2\,\mathtt{expm1}(\alpha_j\ell_1) - \mathtt{expm1}(\alpha_j\ell_2) - \mathtt{expm1}(\alpha_j\ell_3)\right]J^{-\alpha_j/3}(\boldsymbol{C}^{-1}:\mathrm{d}\boldsymbol{E})
\end{aligned} \tag{C.6}$$

To compute $\mathrm{d}\lambda_i$ we differentiate $\boldsymbol{C} = \sum_{i=1}^{3}\lambda_i^2\boldsymbol{N}_i\boldsymbol{N}_i^T$ as

$$\mathrm{d}\boldsymbol{C} = \sum_{i=1}^{3} 2\lambda_i\,\mathrm{d}\lambda_i\boldsymbol{N}_i\boldsymbol{N}_i^T + \lambda_i^2\left(\mathrm{d}\boldsymbol{N}_i\boldsymbol{N}_i^T + \boldsymbol{N}_i\,\mathrm{d}\boldsymbol{N}_i^T\right) \tag{C.7}$$

and used the fact that the eigenvectors are orthonormal i.e., $\langle \boldsymbol{N}_i, \boldsymbol{N}_j \rangle = \delta_{ij}$ so that $\langle \boldsymbol{N}_i, \mathrm{d}\boldsymbol{N}_i \rangle = 0$, we can multiply (C.7) from the left and right by $\boldsymbol{N}_i$

$$\langle \boldsymbol{N}_i, \mathrm{d}\boldsymbol{C}\boldsymbol{N}_i \rangle = 2\lambda_i \, \mathrm{d}\lambda_i \Rightarrow \mathrm{d}\lambda_i = \frac{1}{\lambda_i} \langle \boldsymbol{N}_i, \mathrm{d}\boldsymbol{E}\boldsymbol{N}_i \rangle \tag{C.8}$$

To derive linearization of the eigenvectors, $\mathrm{d}\boldsymbol{N}_i$, we start by differentiating $\boldsymbol{E}\boldsymbol{N}_i = \lambda_i^E \boldsymbol{N}_i$

$$\mathrm{d}\boldsymbol{E}\boldsymbol{N}_i + \boldsymbol{E} \, \mathrm{d}\boldsymbol{N}_i = \mathrm{d}\lambda_i^E \boldsymbol{N}_i + \lambda_i^E \, \mathrm{d}\boldsymbol{N}_i \tag{C.9}$$

taking the inner product of above equation with $\boldsymbol{N}_j$, $j \neq i$ simplifies above equation to

$$\begin{aligned}
&\langle \boldsymbol{N}_j, \mathrm{d}\boldsymbol{E}\boldsymbol{N}_i \rangle + \langle \mathrm{d}\boldsymbol{N}_i, \boldsymbol{E}\boldsymbol{N}_j \rangle = \lambda_i^E \langle \mathrm{d}\boldsymbol{N}_i, \boldsymbol{N}_j \rangle \\
&\langle \boldsymbol{N}_j, \mathrm{d}\boldsymbol{E}\boldsymbol{N}_i \rangle + \beta_j^E \langle \mathrm{d}\boldsymbol{N}_i, \boldsymbol{N}_j \rangle = \lambda_i^E \langle \mathrm{d}\boldsymbol{N}_i, \boldsymbol{N}_j \rangle \\
&\langle \mathrm{d}\boldsymbol{N}_i, \boldsymbol{N}_j \rangle = \frac{1}{\lambda_i^E - \beta_j^E} \langle \mathrm{d}\boldsymbol{E}\boldsymbol{N}_i, \boldsymbol{N}_j \rangle \\
&\mathrm{d}\boldsymbol{N}_i = \sum_{j \neq i} \frac{1}{\lambda_i^E - \beta_j^E} \langle \mathrm{d}\boldsymbol{E}\boldsymbol{N}_i, \boldsymbol{N}_j \rangle \boldsymbol{N}_j
\end{aligned}$$

and finally the linearization of $\ell_i = \log \lambda_i$ is

$$\mathrm{d}\ell_i = \frac{\mathrm{d}\lambda_i}{\lambda_i}. \tag{C.10}$$

which complete $\mathrm{d}\boldsymbol{S}_{\mathrm{iso}}$ for Ogden model.

Note that in the case of two identical eigenvalues, $\lambda_i^E = \beta_j^E$, the denominator of (C.10) becomes zero and leads `NaN` in the computation. In this case we used the following relations [44]

$$\left.\begin{aligned}
\tfrac{\partial \lambda_1^2}{\partial \boldsymbol{C}} &= \boldsymbol{I} - \boldsymbol{N}_3 \boldsymbol{N}_3{}^T \\
\tfrac{\partial \lambda_3^2}{\partial \boldsymbol{C}} &= \boldsymbol{N}_3 \boldsymbol{N}_3{}^T
\end{aligned}\right\} \quad \text{for} \quad \lambda_1 = \lambda_2 \neq \lambda_3 \tag{C.11}$$

$$\frac{\partial \lambda^2}{\partial \boldsymbol{C}} = \sum_{i=1}^{3} \boldsymbol{N}_i \boldsymbol{N}_i^T = \boldsymbol{I} \quad \text{for} \quad \lambda_1 = \lambda_2 = \lambda_3 = \lambda, \tag{C.12}$$

to compute $\mathrm{d}\big(\boldsymbol{N}_i \boldsymbol{N}_i^T\big) = \big(\mathrm{d}\boldsymbol{N}_i \boldsymbol{N}_i^T + \boldsymbol{N}_i \, \mathrm{d}\boldsymbol{N}_i^T\big)$ in (C.5).

Appendix D

Stress linearization in current configuration

As we discussed in section 3.4, for current configuration we need Kirchhoff stress, which can be derived from (C.1)

$$\boldsymbol{\tau} = \boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^T = \boldsymbol{\tau}_{\text{iso}} + \underbrace{\left(k_p JU' - pJ\right)\boldsymbol{I}}_{\boldsymbol{\tau}_{\text{vol}}}, \tag{D.1}$$

and (C.2)

$$\boldsymbol{\tau} = \boldsymbol{F}\boldsymbol{S}\boldsymbol{F}^T = \boldsymbol{\tau}_{\text{iso}} + \underbrace{\left(k_p\hat{U} - p\right)J\hat{U}'\boldsymbol{I}}_{\boldsymbol{\tau}_{\text{vol}}}, \tag{D.2}$$

where the isochoric part is derived in chapter 2 for Neo-Hookean, Mooney-Rivlin and Ogden models.

As we showed in (3.62), for the linearization in current configuration we need to derive $\boldsymbol{F}\,\mathrm{d}\boldsymbol{S}\boldsymbol{F}^T$ term. We derived $\boldsymbol{F}\,\mathrm{d}\boldsymbol{S}_{\text{vol}}\boldsymbol{F}^T$ in (3.71) and $\boldsymbol{F}\,\mathrm{d}\boldsymbol{S}_{\text{iso}}\boldsymbol{F}^T$ for Neo-Hookean in (3.65). Since the volumetric part is the same for different materials, in following we derive $\boldsymbol{F}\,\mathrm{d}\boldsymbol{S}_{\text{iso}}\boldsymbol{F}^T$ for Mooney-Rivlin and Ogden models.

D.1 Mooney-Rivlin model

In this case, multiply (C.3) by $\boldsymbol{F}$ and $\boldsymbol{F}^T$ from left and right gives

$$\begin{aligned}\boldsymbol{F}\,\mathrm{d}\boldsymbol{S}_{\mathrm{iso}}\boldsymbol{F}^T &= -\frac{4}{3}\,\mathrm{trace}(\mathrm{d}\boldsymbol{\epsilon})\left(\mu_1 J^{-2/3}+4\mu_2 J^{-4/3}\right)\boldsymbol{e}_{\mathrm{dev}}\\ &+\frac{2}{3}\left(\mu_1 J^{-2/3}+2\mu_2 J^{-4/3}\right)\left(\mathbb{I}_1\,\mathrm{d}\boldsymbol{\epsilon}-\mathbb{I}_1(\mathrm{d}\boldsymbol{e})\,\boldsymbol{I}\right)\\ &-\frac{8}{3}(\mathrm{trace}(\mathrm{d}\boldsymbol{\epsilon}))\mu_2 J^{-4/3}\left(\mathbb{I}_1(\boldsymbol{e})\boldsymbol{b}-\boldsymbol{b}\boldsymbol{e}\right)\\ &+2\mu_2 J^{-4/3}\left(\mathbb{I}_1(\mathrm{d}\boldsymbol{e})\boldsymbol{b}-\boldsymbol{b}\,\mathrm{d}\boldsymbol{\epsilon}\boldsymbol{b}\right)\\ &+(c_1+c_2)\boldsymbol{I}+\frac{8}{3}\mu_2 J^{-4/3}\left(\mathbb{I}_1(\boldsymbol{e})+2\mathbb{I}_2(\boldsymbol{e})\right)\mathrm{d}\boldsymbol{\epsilon}\end{aligned} \tag{D.3}$$

where

$$\begin{aligned}c_1 &= \frac{16}{9}\mu_2 J^{-4/3}\,\mathrm{trace}(\mathrm{d}\boldsymbol{\epsilon})\left(\mathbb{I}_1(\boldsymbol{e})+2\mathbb{I}_2(\boldsymbol{e})\right).\\ c_2 &= -\frac{4}{3}\mu_2 J^{-4/3}\left(\mathbb{I}_1(\mathrm{d}\boldsymbol{e})+2\mathbb{I}_1(\boldsymbol{e})\mathbb{I}_1(\mathrm{d}\boldsymbol{e})-2\,\mathrm{trace}(\boldsymbol{b}\boldsymbol{e}\,\mathrm{d}\boldsymbol{\epsilon})\right).\end{aligned} \tag{D.4}$$

and we have used

$$\begin{aligned}\boldsymbol{F}\boldsymbol{E}\boldsymbol{F}^T &= \frac{1}{2}\left(\boldsymbol{F}\boldsymbol{C}\boldsymbol{F}^T-\boldsymbol{F}\boldsymbol{F}^T\right)=\boldsymbol{b}\boldsymbol{e}.\\ \mathrm{d}\boldsymbol{E} &= \frac{1}{2}\left(\boldsymbol{F}^T\,\mathrm{d}\boldsymbol{F}\boldsymbol{F}^{-1}\boldsymbol{F}+\boldsymbol{F}^T\boldsymbol{F}^{-T}\,\mathrm{d}\boldsymbol{F}^T\boldsymbol{F}\right)=\boldsymbol{F}^T\,\mathrm{d}\boldsymbol{\epsilon}\boldsymbol{F}.\end{aligned} \tag{D.5}$$

$$\boldsymbol{E}:\mathrm{d}\boldsymbol{E}=\mathrm{trace}(\boldsymbol{E}\,\mathrm{d}\boldsymbol{E})=\mathrm{trace}(\boldsymbol{E}\boldsymbol{F}^T\,\mathrm{d}\epsilon\boldsymbol{F})=\mathrm{trace}(\boldsymbol{F}\boldsymbol{E}\boldsymbol{F}^T\,\mathrm{d}\epsilon)=\mathrm{trace}(\boldsymbol{b}\boldsymbol{e}\,\mathrm{d}\boldsymbol{\epsilon}).$$

D.2 Ogden model

For Ogden model we have

$$\begin{aligned}\boldsymbol{F}\,\mathrm{d}\boldsymbol{S}_{\mathrm{iso}}\boldsymbol{F}^T &= \sum_{i=1}^{3}\lambda_i^2\,\mathrm{d}s_i^{\mathrm{iso}}\boldsymbol{n}_i\boldsymbol{n}_i^T+\lambda_i s_i^{\mathrm{iso}}\left(\boldsymbol{F}\,\mathrm{d}\boldsymbol{N}_i\boldsymbol{n}_i^T+\boldsymbol{n}_i\,\mathrm{d}\boldsymbol{N}_i^T\boldsymbol{F}^T\right)\\ &= \sum_{i=1}^{3}\left(\mathrm{d}\tau_i^{\mathrm{iso}}-2\frac{\mathrm{d}\lambda_i}{\lambda_i}\tau_i^{\mathrm{iso}}\right)\boldsymbol{n}_i\boldsymbol{n}_i^T+\tau_i^{\mathrm{iso}}\boldsymbol{A}_i\end{aligned} \tag{D.6}$$

where for example $\mathrm{d}\tau_1^{\mathrm{iso}}$ can be written

$$\begin{aligned}\mathrm{d}\tau_1^{\mathrm{iso}} &= \sum_{j=1}^{N}\frac{m_j\alpha_j}{3}\left[2\,\mathrm{d}\ell_1\exp(\alpha_j\ell_1)-\mathrm{d}\ell_2\exp(\alpha_j\ell_2)-\mathrm{d}\ell_3\exp(\alpha_j\ell_3)\right]J^{-\alpha_j/3}\\ &-\sum_{j=1}^{N}\frac{m_j\alpha_j}{9}\left[2\,\mathtt{expm1}(\alpha_j\ell_1)-\mathtt{expm1}(\alpha_j\ell_2)-\mathtt{expm1}(\alpha_j\ell_3)\right]J^{-\alpha_j/3}\,\mathrm{trace}\left(\mathrm{d}\boldsymbol{\epsilon}\right)\end{aligned} \tag{D.7}$$

and

$$\boldsymbol{A}_i = 2\frac{\mathrm{d}\lambda_i}{\lambda_i}\boldsymbol{n}_i\boldsymbol{n}_i^T + \mathrm{d}\big(\boldsymbol{n}_i\boldsymbol{n}_i^T\big) + \Big((\nabla_x \,\mathrm{d}\boldsymbol{u})\,\boldsymbol{n}_i\boldsymbol{n}_i^T + \boldsymbol{n}_i\boldsymbol{n}_i^T\,(\nabla_x \,\mathrm{d}\boldsymbol{u})^T\Big)$$

where τ_i^{iso} is defined in (2.50) and we have used $\boldsymbol{F}\boldsymbol{N}_i = \lambda_i\boldsymbol{n}_i$ and $\tau_i^{\mathrm{iso}} = \lambda_i^2 s_i^{\mathrm{iso}}$. Note that the linearization of principal stretches, λ_i, and eigenvectors, $\boldsymbol{n}_i$, are similar to initial configuration but are written in terms of Green-Euler tensor i.e.,

$$\mathrm{d}\lambda_i = \frac{1}{\lambda_i}\langle \boldsymbol{n}_i, \mathrm{d}\boldsymbol{e}\boldsymbol{n}_i\rangle.$$

$$\mathrm{d}\boldsymbol{n}_i = \sum_{j\neq i}\frac{1}{\lambda_i^e - \lambda_j^e}\langle \mathrm{d}\boldsymbol{e}\boldsymbol{n}_i, \boldsymbol{n}_j\rangle\boldsymbol{n}_j.$$

Appendix E

Abaqus Uhyper Subroutine

To simulate a hyperelastic material in Abaqus, we need a user subroutine as described in Abaqus documentation. We follow Abaqus notation and CVEN7511's course notes [76] with the strain energy function as

$$U = U_{vol}(J) + U_{dev}(\bar{\mathbb{I}}_1) = \frac{k}{2}(J-1)^2 + \frac{\mu}{2}\left(\bar{\mathbb{I}}_1 - 3\right) \tag{E.1}$$

where k, μ are bulk and shear moduli, $\bar{\mathbb{I}}_1 = J^{-2/3}\mathbb{I}_1 = J^{-2/3}\,\text{trace}\,\boldsymbol{C}$ is modified first invariant, $\boldsymbol{C} = \boldsymbol{F}^T\boldsymbol{F}$ is right Cauchy-Green tensor and $J = |\boldsymbol{F}|$ is the volume ratio.

To complete the subroutine we need the following terms.

Energy terms:

$$\begin{aligned} U(1) &= U = \frac{k}{2}(J-1)^2 + \frac{\mu}{2}\left(\bar{\mathbb{I}}_1 - 3\right) \\ U(2) &= U_{dev}(\bar{\mathbb{I}}_1) = \frac{\mu}{2}\left(\bar{\mathbb{I}}_1 - 3\right) \end{aligned} \tag{E.2}$$

First derivatives:

$$\begin{aligned} UI1(1) &= \frac{\partial U}{\partial \bar{\mathbb{I}}_1} = \frac{\mu}{2} \\ UI1(2) &= \frac{\partial U}{\partial \bar{\mathbb{I}}_2} = 0 \\ UI1(3) &= \frac{\partial U}{\partial J} = k(J-1) \end{aligned} \tag{E.3}$$

Second derivatives:

$$
\begin{aligned}
UI2(1) &= \frac{\partial^2 U}{\partial \bar{\mathbb{I}}_1^2} = 0 \\
UI2(2) &= \frac{\partial^2 U}{\partial \bar{\mathbb{I}}_2^2} = 0 \\
UI2(3) &= \frac{\partial^2 U}{\partial J^2} = k \\
UI2(4) &= \frac{\partial^2 U}{\partial \bar{\mathbb{I}}_1 \partial \bar{\mathbb{I}}_2} = 0 \\
UI2(5) &= \frac{\partial^2 U}{\partial \bar{\mathbb{I}}_1 \partial J} = 0 \\
UI2(6) &= \frac{\partial^2 U}{\partial \bar{\mathbb{I}}_2 \partial J} = 0
\end{aligned}
\tag{E.4}
$$

Third derivatives:

$$
\begin{aligned}
UI3(1) &= \frac{\partial^3 U}{\partial \bar{\mathbb{I}}_1^2 \partial J} = 0 \\
UI3(2) &= \frac{\partial^3 U}{\partial \bar{\mathbb{I}}_2^2 \partial J} = 0 \\
UI3(3) &= \frac{\partial^3 U}{\partial \bar{\mathbb{I}}_1 \partial \bar{\mathbb{I}}_2 \partial J} = 0 \\
UI3(4) &= \frac{\partial^3 U}{\partial \bar{\mathbb{I}}_1 \partial J^2} = 0 \\
UI3(5) &= \frac{\partial^3 U}{\partial \bar{\mathbb{I}}_2 \partial J^2} = 0 \\
UI3(6) &= \frac{\partial^3 U}{\partial J^3} = 0
\end{aligned}
\tag{E.5}
$$

The following code is provided by the instructor of the CVEN7511 course [76].

Subroutine Code

```
C
      SUBROUTINE UHYPER(BI1,BI2,AJ,U,UI1,UI2,UI3,TEMP,NOEL,
     1 CMNAME,INCMPFLAG,NUMSTATEV,STATEV,NUMFIELDV,FIELDV,
     2 FIELDVINC,NUMPROPS,PROPS)
C
      INCLUDE 'ABA_PARAM.INC'
```

```
C
      CHARACTER*80 CMNAME
      DIMENSION U(2),UI1(3),UI2(6),UI3(6),STATEV(NUMSTATEV),
     1 FIELDV(NUMFIELDV),FIELDVINC(NUMFIELDV),PROPS(NUMPROPS)
C
      PARAMETER (ONE=1.0D0,TWO=2.0D0,THREE=3.0D0,SIX=6.0D0,
     1 ZERO=0.0D0,FOUR=4.0D0,SEVEN=7.0D0,TWSEV=27.0D0,
     2 HALF=0.5D0,NINE=9.0D0)
C
C..Incompressibility flag defined to be
C..1 if the material is specified as incompressible, or
C..0 if the material is specified as compressible.
C
      INCMPFLAG=0
C
C -------------------------------------------------------------
C
C     PROPS(1) - E
C     PROPS(2) - 0.5-NU
C
      EYOUNG=PROPS(1)
      ENU=HALF-PROPS(2)
      DMU=EYOUNG/(TWO*(ONE+ENU))
      DKAPPA=EYOUNG/(THREE*(ONE-TWO*ENU))
C       WRITE(*,*) 'DKAPPA = ',DKAPPA
C       WRITE(*,*) 'DMU = ',DMU
C
```

```
C -------------------------------------------------------------
C
C   BI1 = \bar{I}_1
C   BI2 = \bar{I}_2
C   AJ  = J = det(F)
C
C -------------------------------------------------------------
C
C...ASSIGN ISVs FOR VIEWING
STATEV(1)=AJ
STATEV(2)=BI1
C
C...FREE ENERGY
C
C...FULL TERM, VOLUMETRIC AND ISOCHORIC
U(1)=HALF*DKAPPA*(AJ-ONE)**TWO+HALF*DMU*(BI1-THREE)
C
C...ISOCHORIC ONLY
U(2)=HALF*DMU*(BI1-THREE)
C
C...DERIVATIVES
C
UI1(1)=HALF*DMU
UI1(2)=ZERO
UI1(3)=DKAPPA*(AJ-ONE)
C
UI2(1)=ZERO
```

```
UI2(2)=ZERO

UI2(3)=DKAPPA

UI2(4)=ZERO

UI2(5)=ZERO

UI2(6)=ZERO

C

UI3(1)=ZERO

UI3(2)=ZERO

UI3(3)=ZERO

UI3(4)=ZERO

UI3(5)=ZERO

UI3(6)=ZERO

C

RETURN

END

C
```